

## THESIS / THÈSE

### MASTER EN SCIENCES INFORMATIQUES

#### Modèle de réseau de files d'attente pour la navigation aérienne

Mimbe, Benis Thierry

*Award date:*  
2000

*Awarding institution:*  
Université de Namur

[Link to publication](#)

#### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

#### Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



FUNDP

INSTITUT D'INFORMATIQUE

**Modèle de réseau de files d'attente  
pour la navigation aérienne**

**MIMBE Benis Thierry**

Mémoire présenté en vue  
de l'obtention du diplôme  
de licencié en informatique

Promoteur : M. Noirhomme

RUE GRANDGAGNAGE, 21 • B – 5000 NAMUR (BELGIUM)



## Abstract

Our object of study is a queueing network which consists of a collection of (connected) single server stations. Exogenous arrivals into the system form independent Poisson streams, service times are exponential and we have Markovian routing of customers between stations. Recent results concerning linear programming (LP) based approaches enable us to establish a simple and intuitive stability condition. This is of interest in its own right, but also enables us to progress with a study of optimal scheduling and performance analysis. Our methodology here is also based on LP. A primal-dual approach exploits the fact that the system satisfies (approximate) conservation laws to yield performance guarantees for a natural index-based scheduling heuristic. We are also able to analyse the performance of an arbitrary priority policy.

L'objet de ce travail est de proposer un réseau de files d'attente comportant plusieurs classes d'avions. Dans la littérature, ces modèles sont appelés *Multiclass Queueing Networks*. Ces réseaux comprennent un ensemble de stations de services interconnectées. Dans ce modèle, le schéma d'arrivée des clients venant de l'extérieur est un flux indépendant de *Poisson*. De plus, les temps de service au niveau des stations sont exponentiels et le routage des clients entre les stations est markovien. L'approche de *Programmation Linéaire* (LP) permet d'établir une condition de stabilité simple et intuitive, ce qui rend possible une étude plus approfondie des questions d'ordonnancement optimal et une analyse des performances. La méthodologie exposée se base également sur la méthode de *programmation linéaire*. L'approche "primal-dual" repose sur le fait que le système satisfait aux lois de conservation afin de fournir des garanties de performance pour une heuristique d'ordonnancement basée sur un index. En outre, il sera possible d'analyser la performance d'un système de priorité arbitraire.

## Tables des matières

Introduction .....	1
Chapitre 1 .....	4
Problématique de l'augmentation du trafic aérien .....	4
1.1 Les contrôleurs aériens .....	5
1.2 Les pilotes et les ingénieurs .....	6
1.3 Intérêts contradictoires .....	7
Chapitre 2 .....	10
Modèles de file d'attente .....	10
2.1 Processus markoviens discrets .....	10
2.2 Chaîne de Markov homogène .....	12
2.2.1 Matrice de transition .....	12
2.2.2 Matrice stochastique .....	12
2.3 Équations de Chapman-Kolmogorov .....	13
2.4 Processus de naissance et de mort .....	15
2.5 Files d'attente .....	15
2.5.1 Schéma d'arrivée .....	16
2.5.2 Le processus de service .....	16
2.5.3 Stratégie de gestion de la file .....	17
2.6 Processus de Poisson et ses propriétés .....	18
2.7 Distribution de service exponentielle .....	20
2.8 Notations de Kendall .....	20
2.9 Mesures de performance d'une file d'attente .....	21
2.10 Formule de Little: $L = \lambda W$ .....	23
2.11 Architecture générale du modèle .....	23
2.11.1 La porte (GATE) .....	24
2.11.2 Le taxiway (TWY) .....	27
2.11.3 La piste (RWY) .....	28
2.11.4 L'espace aérien de gestion de l'aéroport (TMA) .....	33
2.11.5 Le secteur (SEC) .....	33
Chapitre 3 .....	35
Réseaux de files d'attente .....	35
3.1 Généralités .....	35
3.2 Réseau de files d'attente pour la navigation aérienne .....	37
Chapitre 4 .....	39
L'analyse de la stabilité, de l'optimisation et de la performance des réseaux de files d'attente multiclassées de Markov par une approche de programmation linéaire .....	39
4.1. Modèle MQNET .....	41
4.2 Stabilité du modèle MQNET .....	44
4.3 Optimisation approximative des règles d'index par les lois de conservation approximatives .....	48
4.4 Analyse de performance des règles de priorité par les lois de conservation approximatives .....	57
Chapitre 5 .....	62
MQNET: étude du cas de la navigation aérienne .....	62
5.1 Le modèle MQNET de la navigation aérienne .....	62
5.2 Stabilité du modèle MQNET .....	76
5.3 L'optimisation approximative des règles d'index par les lois de conservation approximatives .....	83
5.4 Analyse de performance des règles de priorité par les lois de conservation approximatives .....	96
Chapitre 6 .....	102
Implémentation du modèle MQNET .....	102
Définition des types .....	102
Définition des constantes .....	103
Définition des procédures et des fonctions .....	104
Bibliographie .....	108
Annexes .....	110



## Introduction

Lorsqu'il faudra choisir un objet symbole du XX<sup>ème</sup> siècle, les futurs historiens ne prendront pas grand risque en optant pour l'avion. Si au début du siècle, les évolutions hésitantes de quelques pionniers soulevaient l'admiration des foules, nous n'hésitons guère, quelques générations plus tard, à confier notre vie à l'aviation civile mondiale. Soumis à une croissance exponentielle, ce grand système technique peut se targuer d'un taux d'accidents remarquablement faible. Mais serait-il en fin de cycle? Des signes d'essoufflement apparaissent.

Le trafic aérien a connu une progression très importante pendant les dernières décennies et toutes les prévisions tendent à montrer que cette croissance ne devrait guère se ralentir dans les prochaines années. De 1986 à 1996, le trafic a connu un accroissement de 66% et on prévoit pour 2006 un accroissement par rapport à 1996 de l'ordre de 35% à 70% (Thomas *et alii* 1997).

Dans ces conditions, la congestion du ciel est en passe de devenir le facteur limitant de la croissance du trafic aérien, du moins en Europe, et la pression des compagnies aériennes se fait de plus en plus forte pour que la gestion du trafic soit à la fois plus souple et plus efficace.

Sur les tableaux d'affichage des aéroports, les indications des vols retardés s'accumulent un peu trop au goût des passagers et des compagnies aériennes. Dans l'obscurité des salles de contrôle, l'inquiétude gagne parfois ces acteurs méconnus qui, quotidiennement, sont responsables de la sécurité de milliers de vies humaines.

Parallèlement, les avions sont de plus en plus performants. Leur structure, leurs moteurs et leur cockpit sont conçus, fabriqués et équipés avec les outils techniques les plus sophistiqués (FMS<sup>1</sup>, GPS<sup>2</sup>, Data Link<sup>3</sup>) qui devraient permettre, à plus ou moins long terme, de changer radicalement les techniques de contrôle des aéronefs.

---

<sup>1</sup> Flight Management System: calculateur assurant du vol des avions. Il s'agit de la version moderne de ce qu'on appelle "le pilote automatique".

<sup>2</sup> Global Positioning System: Système de navigation par satellite

C'est dans ce cadre qu'est aujourd'hui posé dans les compagnies le concept de Free Flight (vol sans contrainte), un concept qui tendrait, sous certaines conditions, à affranchir les avions du contrôle aérien tel qu'il est pratiqué aujourd'hui et qui permettrait ainsi aux aéronefs de suivre le cheminement de leur choix dans certaines zones de l'espace aérien.

Bien que le chemin à parcourir soit encore long avant la mise en place d'un tel concept, il apparaît crucial de disposer d'un outil de planification à long terme afin de pouvoir donner ou refuser l'autorisation à certains aéroports d'ouvrir telles ou telles lignes ou d'accroître le nombre de vols pour une destination précise.

Alors que certains modèles permettent de gérer le trafic existant à un moment donné, il semble qu'il n'existe pas encore de modèle qui prenne en compte toutes les étapes d'un vol, depuis la porte d'embarquement dans l'aéroport de départ jusqu'à la porte de débarquement de l'aéroport d'arrivée. Dans ce contexte, la firme Eurocontrol, la société européenne chargée du trafic aérien en Europe, a commandé une étude visant à proposer un modèle stochastique global qui prend en compte toutes les étapes d'un vol.

Ce mémoire s'inscrit dans le cadre de cette recherche. Au cours de ce travail, nous n'aurons pas la prétention de répondre à toutes les questions relatives à cette problématique, mais nous tâcherons d'apporter quelques pistes de réflexion permettant de faire progresser les recherches dans ce domaine. Notre objectif sera de proposer un analytique capable de:

- prédire le temps moyen d'un vol et, en particulier, les différents temps d'attente;
- de détecter les serveurs qui limitent l'efficacité du système ATM<sup>4</sup> dans son entièreté pour une demande donnée;
- de prédire les taux d'occupation des différents serveurs.

Notre travail comprend six chapitres. Le premier chapitre est consacré à la présentation des différents acteurs du trafic aérien. Dans cette partie, nous expliquons en quoi le problème de la saturation de l'espace aérien constitue non seulement un problème technique mais aussi un problème social et institutionnel.

<sup>3</sup> Liaison de données numériques entre l'avion et le sol ou entre les avions permettant le transfert de paramètres complexes (altitude, position, vitesse).

<sup>4</sup> Air Traffic Management



Dans le second chapitre, nous analysons tour à tour chacune des étapes d'un vol à savoir le système de portes (Gates), le taxiway (TWY), la piste (RWY) l'espace aérien de gestion de l'aéroport (TMA) et les secteurs. D'abord, nous proposons un aperçu théorique des divers concepts permettant d'établir les modèles de files d'attente. Ensuite, nous présentons un modèle de file d'attente pour chacune des étapes du vol.

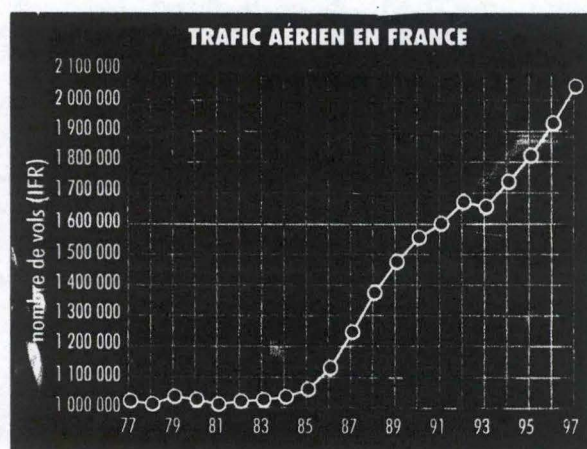
Dans le troisième chapitre, nous tentons d'intégrer au sein d'un modèle global de réseau de files d'attente toutes les étapes d'un vol, de la porte d'embarquement d'un aéroport de départ à la porte de débarquement d'un aéroport d'arrivée.

Le quatrième chapitre figure comme la pièce maîtresse de ce mémoire. En effet, elle présente un modèle de réseau de files d'attente desservant plusieurs classes de clients. Ce modèle théorique est appliqué dans le cinquième chapitre au cas de la navigation aérienne. Enfin, le sixième chapitre propose une implémentation du modèle.

## Chapitre 1

# Problématique de l'augmentation du trafic aérien

Le transport aérien a connu en Europe un extraordinaire développement depuis les débuts des années 1980. Pour s'en convaincre, il suffit de regarder la croissance exponentielle du nombre des vols. Chaque année, le trafic augmente en moyenne de 4% (en 1997, l'augmentation a atteint un niveau record de 7%), ce qui représente un doublement du trafic tous les dix ans (Figure 1.1)! Les origines de ce développement commercial sont faciles à identifier: nouvelles destinations, vols plus fréquents, tarifs plus bas grâce à l'introduction de la concurrence de lignes autrefois monopolisées par une compagnie (Stuchlik 1999).

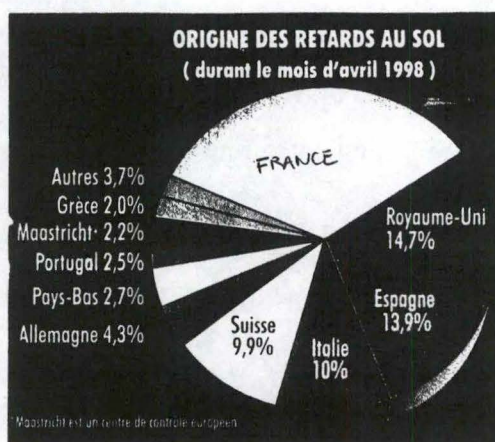


**Fig. 1.1-** En 1997, l'augmentation du trafic a atteint un niveau record.(Stuchlik 1999)

Cependant, ce qu'on pourrait appeler les "infrastructures" ont du mal à suivre cette croissance effrénée du trafic. En effet, l'espace aérien européen est en permanence à la limite de la saturation aux heures de pointe. Saturation au voisinage des aéroports mais, particularité européenne, saturation aussi et surtout sur les routes aériennes entre les aéroports. La principale manifestation de ce phénomène est le nombre croissant des retards des vols. Il n'est pas rare aujourd'hui de voir son avion retardé d'une demie heure ou plus (figure 2.2), situation qui risque de devenir intolérable à terme pour les compagnies aériennes. Pour ces dernières, les enjeux sont de taille: elles ont encaissé 350 milliards de francs de recette en 1996, et



espèrent dépasser les 700 milliards en 2010. Le coût des retards était estimé à 2 milliards et demi de francs français en 1996, un chiffre qui ne prend pas en compte les inconvénients pour les passagers. Enfin, le montant des redevances de contrôle aérien que doivent payer les compagnies aériennes dépasse le 20 milliards de francs, ce qui constitue un montant plus élevé que les frais en carburant!



**Fig. 1. 2-** Les retards sont une mesure de la performance du trafic arien. (Stuchlik 1999)

## 1.1 Les contrôleurs aériens

Le contrôleur aérien est un métier méconnu du grand public. Quand les médias parlent des aiguilleurs du ciel, c'est souvent pour annoncer une grève qui va clouer les avions au sol. Pour ceux qui prennent l'avion, le terme de contrôle aérien est plus familier: c'est celui qu'invoque le commandant pour expliquer le retard au décollage. Bref, le contrôle aérien (ATC, Air Traffic Control) reste dans l'ombre, et s'il en sort momentanément, ce n'est pas à son avantage.

Le contrôleur aérien a une lourde responsabilité. Il surveille une zone d'espace aérien, appelée le secteur, et guide les avions qui la traversent. Il doit s'assurer que ceux-ci ne s'approchent pas trop les uns des autres en violant les normes de séparation (de 9 à 18 km dans le plan horizontal, de 300 m à 600 m dans le plan vertical) (Alliot 1998). Pour ce faire, il donne des instructions au pilote, ou plutôt il négocie avec ce dernier la route qu'il prendra. C'est un métier qui nécessite de l'agilité intellectuelle, de la rapidité, une bonne résistance au stress et des facultés d'adaptation. Selon l'expression consacrée, le contrôle aérien est un "artisanat intellectuel". Par exemple, au début des années 1990, quand les contrôleurs français



recevaient un avion russe dans leur secteur, ils faisaient s'écarter tous les autres, de peur que le pilote russe comprenne mal leurs instructions en anglais!

Le ciel européen est donc saturé. Or, qui parle de saturation parle de capacité. Quelle est donc cette capacité de l'espace aérien? Comment la définit-on? En théorie, c'est le nombre maximal d'avions que l'on peut faire circuler simultanément dans le ciel sans qu'il y ait des collisions entre eux. Éviter que les avions entrent en collision, c'est la mission essentielle du contrôle aérien, et c'est la performance du contrôle aérien qui va définir la capacité de l'espace aérien d'un pays. Mais cette vision fait-elle l'unanimité?

Pour les contrôleurs, la performance du contrôle aérien renvoie effectivement à la sécurité. En effet, le contrôleur est en première ligne en cas de collision, car il est le seul responsable de la vie du pilote et celle des passagers. Pensons que si un contrôleur peut gérer une vingtaine d'avions aux heures de pointe, il a entre les mains la vie de plusieurs milliers de personnes. En cas d'accident, il devra faire face non seulement au tribunal, puisqu'il est responsable juridiquement, mais aussi à un profond traumatisme psychologique. Le contrôleur arrive notamment grâce au groupe à supporter la hantise de l'accident et à assumer cette lourde responsabilité et le stress qui l'accompagne.

## 1.2 Les pilotes et les ingénieurs

L'interlocuteur direct du contrôleur est le pilote, qui représente les intérêts de sa compagnie aérienne. Or, pour celui-ci, la performance du contrôle aérien correspond à d'autres facteurs. Certes, la sécurité est toujours primordiale, mais elle est considérée comme un acquis élémentaire. Ce qui importe à la compagnie aérienne et donc au pilote, c'est la ponctualité du vol. Un vol en retard signifie des heures supplémentaires à payer, du carburant consommé en plus, voire des nuits d'hôtel à rembourser aux passagers. Les compagnies insistent donc sur la ponctualité et sur le tracé direct des vols. Ensuite vient le montant des redevances de contrôle, que les compagnies ont vu monter sensiblement ces dernières années, d'une part en raison de l'augmentation du prix du contrôle aérien et d'autre part à cause de l'ouverture de nouvelles lignes.

Un troisième protagoniste est l'ingénieur. Celui-ci conçoit et développe les systèmes qui seront utilisés directement ou indirectement par le contrôleur: radars, écrans, dispositifs de régulation du trafic, etc. Beaucoup d'attentes reposent sur l'ingénieur, car on compte sur ses



systèmes pour résoudre le problème de la saturation de l'espace aérien. Même si la performance du contrôle aérien est autant un problème social et institutionnel que technique, on préfère recourir à l'ingénieur en attendant de lui la solution miracle. L'ingénieur, lui, tendra plus à suivre ses propres critères. Un contrôle aérien performant signifie, selon lui, un contrôle aérien à la pointe de la technique. Par exemple, certains ingénieurs s'indignent qu'en l'an 2000 les contrôleurs utilisent encore des strips (Mackay *et alii* 1999), il serait bien plus judicieux de remplacer ces bouts de papiers surannés par des strips virtuels, électroniques que l'on déplacerait à l'aide d'une souris ou d'un joystick. Mais au grand dam des ingénieurs, les contrôleurs se montrent réticents à cette volonté de progrès. Les compagnies ne voient pas non plus les ingénieurs d'un bon œil, car les systèmes à la pointe de la technique coûtent souvent chers, ce qui augmente les redevances.

Le panorama du contrôle aérien serait incomplet, si on ne mentionnait pas l'administration d'État. En effet, le contrôle aérien dépend plus ou moins directement de l'État dans tous les pays d'Europe et ce dernier poursuit ses propres objectifs, principalement tournés vers la politique intérieure. Il faut minimiser les mouvements de grève, car une poignée de contrôleurs peut paralyser un pays.

### 1.3 Intérêts contradictoires

Le contrôle aérien performant sera celui "qui ne fait pas de vagues", en d'autres termes qui ne générera ni grèves ni accidents. Plusieurs stratégies vis-à-vis des contrôleurs sont possibles, allant de la suppression du droit de grève (cas de l'Allemagne avant 1992) à la politique d'autogestion des contrôleurs qui s'est instaurée de fait en France. En outre, l'Etat poussera le contrôle aérien à s'équiper chez un industriel national (Thomson en France, Indra en Espagne, etc.). On voit donc que le contrôle aérien est au centre d'intérêts convergents ou contradictoires entre divers acteurs, qui ont des visions différentes de la performance: sécurité bien sûr, mais aussi capacité, ponctualité, longueur des trajets, prix, modernité des équipements, absence de conflits sociaux, politique industrielle, etc. Nous allons ici choisir de nous attacher à une dimension particulière, celle de la capacité, car elle joue le rôle de ressource critique. Si l'on a suffisamment de capacité, la sécurité devrait être assurée et la ponctualité aussi. Or, aujourd'hui, la capacité manque.

Certes, au fur et à mesure de l'augmentation du trafic, les différents pays ont réussi à absorber les vols supplémentaires, en accroissant cette capacité par différents moyens:



amélioration de la précision des radars, diminution des séparations entre les avions, augmentation des effectifs de contrôleurs, etc. Cependant, il devient de plus en plus difficile d'augmenter cette capacité, et on risque bientôt de se heurter à une limite absolue: c'est ce qu'on appelle "le mur de la capacité". Que se passera-t-il quand cette limite sera atteinte? Trois grandes représentations se dégagent du problème du "mur de capacité". Ces représentations sont appelées la mort, le pourrissement et le chaos[Stuchlik 1999].

La mort correspond à une vision catastrophique de l'atteinte du mur de la capacité. La croissance du trafic aérien créerait une avalanche d'accidents. La saturation de l'espace aérien serait telle que le nombre de collisions entre avions finirait par dissuader les passagers et entraînerait la mort de l'industrie du transport aérien. Les contrôleurs parlent d'accidents tellement graves qu'ils mettraient en péril l'existence du trafic aérien, un peu comme si "le ciel leur tombait sur la tête".

Si cette première représentation se caractérise par l'idée de rupture, celle du pourrissement évoque au contraire la continuité. Dans celle-ci, le mur de capacité n'est pas vu comme une limite en soi, mais comme un processus de dégradation progressive. Cette représentation fréquente chez les ingénieurs s'appuie sur l'hypothèse suivante: "une croissance du trafic aérien de 1% entraîne une augmentation des retards de 5%". Cette assertion provient des estimations faites par le CFMU<sup>5</sup>, l'organisme qui donne les autorisations de décollage sur toute l'Europe. Le pourrissement a l'avantage de dédramatiser relativement la situation: plus de rupture, mais une croissance qui donne l'illusion d'être maîtrisé. En d'autres termes, "ça empire, mais ça reste pareil, c'est toujours la même chose". L'estimation fournit un modèle linéaire de la saturation de l'espace aérien, ce qui donne l'impression que la situation reste sous contrôle.

Le chaos remet en question l'hypothèse de linéarité entre le trafic et les retards. En effet, si le transport aérien était resté une affaire de lignes aériennes de point à point, les retards n'auraient pas d'incidence d'une ligne aérienne sur une autre. Mais les compagnies se sont organisées en *hubs*, c'est-à-dire en correspondance autour d'aéroports particuliers. Ainsi, Air France effectue toutes ses correspondances autour du *hub* de Roissy alors que Lufthansa a choisi Francfort. Or, à partir du moment où les lignes aériennes deviennent interdépendantes par le biais des *hubs*, des effets de cascade, voire d'avalanche, peuvent se produire.

---

<sup>5</sup> CFMU: Central Flow Management Unit (cellule de gestion centralisée des flux)



Si une faible perturbation du système a des conséquences limitées, il semble qu'au delà d'un certain seuil le système peut se mettre à errer complètement, et se trouver désorganisé d'une telle façon que l'on ne sache plus comment agir dessus pour revenir à la normale. Le système est donc devenu chaotique.

A l'appui de cette hypothèse, voici ce que dit un spécialiste des questions ATC à Air France: "Durant l'été dernier, il y a eu un jour où notre *hub* de Roissy a explosé. Les retards étaient tels que l'on ne savait plus s'il fallait attendre l'arrivée de tel vol pour faire décoller tel autre, s'il fallait assurer la correspondance, si le créneau de décollage accordé dans tel aéroport permettrait d'accrocher avec tel autre. Le *hub* était complètement désorganisé, c'était la panique. Personne ne savait plus quoi faire, on était complètement perdu."

Le chaos semble le scénario le plus probable. Le système du transport aérien fonctionnerait à peu près normalement la majeure partie du temps, mais aurait des "pointes chaotiques" où il se déréglerait totalement et de façon difficilement prévisible, un peu à la manière des bouchons sur les routes, qui surviennent parfois sans que l'on sache pourquoi. La question qui se pose maintenant est de savoir comment retarder cette échéance, voire l'éviter, si c'est possible. Peut-on encore reculer le mur de capacité?

## Chapitre 2

### Modèles de file d'attente

#### 2.1 Processus markoviens discrets

Très souvent, quand on étudie un phénomène qui dépend du hasard, il y a lieu de prendre en compte l'évolution de ce phénomène au cours du temps: le modèle mathématique correspondant est appelé "processus stochastique", ou plus simplement "processus".

En fait, de même que l'expression de "variable aléatoire", il y a plusieurs significations au terme de "processus stochastique". Nous n'expliquons pas ces diverses définitions possibles qui varient selon le niveau où l'on se situe et selon les problèmes à résoudre.

Disons seulement que, si  $T$  représente le temps, un processus  $(X_t)_{t \in T}$  est une famille de variables aléatoires indexées par  $T$ . Pour tout élément  $t$  de  $T$ ,  $X_t$  correspond à l'état du processus à l'instant  $t$ . Le temps  $T$  est un intervalle de  $N$  (cas discret) ou de  $R$  (cas continu).

Soit  $E$  l'ensemble des états possibles du processus à chaque instant  $t$ ; conformément aux conventions introduites précédemment, si  $e$  est un élément de  $E$ ,  $\Pr[X_t = e]$  est la probabilité que le processus  $X$  soit à l'état  $e$  à l'instant  $t$ .

De même si  $i$  et  $j$  sont deux éléments de  $E$ ,  $\Pr[X_{t+h} = j \mid X_t = i]$  est la probabilité que le processus  $X$  soit dans l'état  $j$  à l'instant  $t+h$  sachant qu'il était à l'état  $i$  à l'instant  $t$ .

La loi conjointe d'un processus  $(X_t)_{t \in T}$  est caractérisée par la famille des lois conjointes  $(X_{t_1}, \dots, X_{t_n})$  où  $(t_1, \dots, t_n)$  parcourt l'ensemble des séquences croissantes finies d'éléments de  $T$ .

Quand on le peut, on choisit un ensemble d'états  $E$  suffisamment riche pour que l'évolution du processus à partir d'un instant  $t$  ne dépende que de l'état  $i$  de ce processus à



l'instant  $t$ , et ceci indépendamment de la façon dont l'état  $i$  a été atteint. Dans ce cas, on dit que le processus  $X$  est markovien.

Autrement dit, un processus  $(X_t)_{t \in T}$  est markovien si le futur est indépendant du passé sachant le présent. Si l'ensemble  $E$  des états est fini ou dénombrable (cas discret) au niveau du système de probabilités conjointes, cela signifie que quelle que soit la séquence croissante  $(t(1), \dots, t(m), \dots, t(n))$  d'éléments de  $T$ , et pour toute séquence associée  $(i_1, \dots, i_n)$  d'éléments de  $E$ , on obtient le résultat suivant:

$$\Pr[X_{t(m+1)} = i_{m+1}, \dots, X_{t(n)} = i_n \mid X_{t(m)} = i_m] =$$

$$\Pr[X_{t(m+1)} = i_{m+1}, \dots, X_{t(n)} = i_n \mid X_{t(1)} = i_1, \dots, X_{t(m)} = i_m]$$

Si le processus  $X$  est markovien et si l'ensemble  $E$  des états est fini ou dénombrable, la loi du processus sur l'intervalle  $T = \mathbb{R}^+$  est caractérisée par la loi initiale (c'est-à-dire la loi de  $X_0$ ) et par le "semi-groupe de transition" c'est-à-dire par la loi de  $X_t$  sachant  $X_s$  pour tout couple  $(s, t)$  avec  $s < t$ . En effet, pour toute séquence associée  $(i_1, \dots, i_n)$  d'éléments de  $E$ , on a:

$$\Pr[X_{t(1)} = i_1, \dots, X_{t(n)} = i_n] = \Pr[X_{t(1)} = i_1] \prod_{k=2}^n \Pr[X_{t(k)} = i_k \mid X_{t(k-1)} = i_{k-1}]$$

et 
$$\Pr[X_{t(1)} = i_1] = \sum_{i \in E} \Pr[X_0 = i] \Pr[X_{t(1)} = i_1 \mid X_0 = i]$$

Cette propriété s'étend au cas où l'ensemble des états est "continu", mais la formalisation en est beaucoup plus compliquée.

On dit que le processus markovien discret  $X$  est homogène si, quel que soit  $(s, t, i, j)$  élément de  $(\mathbb{R}^+ \times T \times E \times E)$ :

$$\Pr[X_t = i \mid X_0 = j] = \Pr[X_{s+t} = i \mid X_s = j]$$

Si l'ensemble des temps est un intervalle de  $\mathbb{N}$  (éventuellement  $\mathbb{N}$  tout entier), on dit qu'il s'agit d'une "chaîne de Markov".

## 2.2 Chaîne de Markov homogène

### 2.2.1 Matrice de transition

Rappelons d'abord qu'une chaîne de Markov homogène est un processus de Markov  $X$  dont l'ensemble  $T$  des temps est un intervalle de l'ensemble des entiers.

On appelle matrice de transition la famille  $(\theta_{ij})_{(i,j) \in E \times E}$  définie par  $\theta_{ij}(k) = \Pr[X_{k+1} = j | X_k = i]$ .

Si la chaîne de Markov est homogène, ce que nous supposons désormais,  $\theta$  ne dépend pas de  $k$ . Si l'ensemble des états  $E$  est un ensemble fini, soit  $\text{card}(E) = n$ ,  $\theta$  est une matrice au sens usuel, dès qu'on a identifié  $E$  et l'ensemble  $\{1, \dots, n\}$ .

Cette matrice caractérise le semi-groupe de transition; plus précisément, dans le cas ci-dessus – cas d'une chaîne de Markov homogène et  $E = \{1, \dots, n\}$  – on a:

$$\Pr[X_k = i | X_0 = j] = (\theta^k)_{ij} \quad (2.1)$$

où  $(\theta^k)_{ij}$  désigne le terme de la  $i^{\text{ème}}$  ligne et de la  $j^{\text{ème}}$  colonne de la matrice  $\theta^k$  (avec les règles de manipulation usuelles pour les matrices). En effet, il suffit de vérifier cette propriété en raisonnant par récurrence sur  $k$ . Supposons donc que la relation (2.1) soit satisfaite, on a:

$$\begin{aligned} \Pr[X_{k+1} = i | X_0 = j] &= \sum_{e=1}^n \Pr[X_{k+1} = i | X_k = e] \Pr[X_k = e | X_0 = j] \\ &= (\theta * \theta^k)_{ij} = (\theta^{k+1})_{ij} \end{aligned}$$

Ce qui achève le raisonnement par récurrence.

### 2.2.2 Matrice stochastique

Soit  $\theta$  une matrice carrée ( $n \times n$ ) dont tous les termes sont positifs. On dit que  $\theta$  est une matrice stochastique si quel que soit  $i$ ,  $\sum_{j=1}^n \theta_{ij} = 1$ . En d'autres termes, la somme des éléments d'une ligne de la matrice stochastique vaut 1.



Si  $\theta$  est une matrice stochastique, on vérifie immédiatement, en raisonnant par récurrence sur  $k$ , que  $\theta^k$  est aussi une matrice stochastique.

Toute matrice de transition est une matrice stochastique. Réciproquement, toute matrice stochastique peut être considérée comme une matrice de transition.

## 2.3 Équations de Chapman-Kolmogorov

Dans la suite de ce chapitre, on va se limiter à l'étude de processus markoviens homogènes dont l'ensemble  $E$  des états est fini ou dénombrable et qui évoluent en temps continu.

Pour tout couple  $(i, j)$  d'éléments de  $E$ , la probabilité de passer de l'état  $i$  à l'état  $j$  pendant  $t+dt$  est donnée par la formule suivante:

$$P_{ij}(t+dt) = \sum_k P_{ik}(t)P_{kj}(dt) \quad (2.2)$$

Pour tout état  $k$  différent de  $j$ , nous avons:

$$\frac{dP_{kj}}{dt} = \lim_{dt \rightarrow 0} \frac{P_{kj}(t+dt) - P_{kj}(t)}{dt} \quad (2.3)$$

Lorsque  $t=0$ , l'équation précédente (2.3) devient:

$$\left. \frac{dP_{kj}}{dt} \right|_{t=0} = \lim_{dt \rightarrow 0} \frac{P_{kj}(dt)}{dt}$$

En posant  $\left. \frac{dP_{kj}}{dt} \right|_{t=0} = q_{kj}$ , nous obtenons la relation suivante:

$$P_{kj}(dt) = q_{kj}dt + O(dt) \quad (2.4)$$

Si  $k = j$ , alors nous avons

$$\frac{dP_{jj}}{dt} = \lim_{dt \rightarrow 0} \frac{P_{jj}(t+dt) - P_{jj}(t)}{dt} \quad (2.5)$$

Lorsque  $t=0$ , l'équation précédente (2.5) devient:

$$\left. \frac{dP_{ij}}{dt} \right|_{t=0} = \lim_{dt \rightarrow 0} \frac{P_{ij}(dt)}{dt}$$

En posant  $\left. \frac{dP_{ij}}{dt} \right|_{t=0} = q_{ij}$ , nous obtenons la relation suivante:

$$P_{ij}(dt) = 1 - q_{ij}dt + O(dt) \quad (2.6)$$

En injectant les relations (2.4) et (2.6) dans (2.2), nous obtenons:

$$P_{ij}(t + dt) = \sum_{k \neq j} P_{ik}(t)(q_{kj}dt + O(dt)) + P_{ij}(t)(1 - q_{ij}dt + O(dt))$$

qui équivaut à :

$$P_{ij}(t + dt) - P_{ij}(t) = \sum_{k \neq j} P_{ik}(t)q_{kj}dt - P_{ij}(t)q_{ij}dt + O(dt) \quad (2.7)$$

En divisant membre à membre l'équation précédente (2.7) par  $dt$  et en prenant la limite lorsque  $dt \rightarrow 0$ , on obtient:

$$\frac{dP_{ij}(t)}{dt} = \sum_{k \neq j} P_{ik}(t)q_{kj} - P_{ij}(t)q_{ij}$$

Ce qui implique l'équation suivante:

$$\frac{dP_j(t)}{dt} = \sum_{k \neq j} P_k(t)q_{kj} - P_j(t)q_{jj}$$

En passant à la limite lorsque  $t \rightarrow \infty$ , il vient:

$$\sum_{k \neq j} P_k(t)q_{kj} - P_j(t)q_{jj} = 0$$

Ces équations sont appelées "équations de balance". Cette relation est fondamentale dans l'analyse des modèles de files d'attente.

L'adjonction de la condition  $\sum_j P_j = 1$  permet d'obtenir une solution.



## 2.4 Processus de naissance et de mort

Parmi la classe des processus continus de Markov, il y a le cas spécial du processus de naissance et de mort (Birth-Death process) dans lequel les états du système changent de tout au plus 1 (que ce soit en plus ou en moins) dans chaque intervalle de temps infinitésimal. Dans de tels cas, nous parlons de taux de naissance  $\lambda_k$ , qui est le taux moyen de naissance quand le système contient  $k$  clients, et du taux de mort  $\mu_k$ , qui est le taux moyen par lequel la population diminue quand elle est de taille  $k$ .

Partant de l'équation de balance (2.8), on peut démontrer que la probabilité d'avoir  $k$  clients dans le système est donnée par la relation suivante:

$$P_k = P_0 \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} \quad (2.9)$$

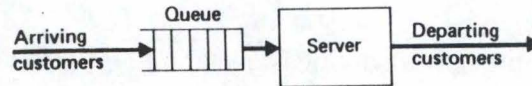
où

$$P_0 = \frac{1}{1 + \sum_{k=1}^{\infty} \prod_{i=0}^{k-1} \lambda_i / \mu_{i+1}} \quad (2.9')$$

L'application de cette solution d'équilibre nous amène directement à la classe des systèmes de files d'attente élémentaires.

## 2.5 Files d'attente

Le comportement de base d'un système de file d'attente est représenté à la Figure 2.1. L'unité opérationnelle qui traverse le système est appelée "client". Si un client souhaite utiliser le serveur, mais le trouve occupé, il doit attendre son tour sur la file d'attente pour être servi. À un certain moment, un client de la file est choisi en fonction d'une règle définie appelée "discipline de service". Une fois servi, le client quitte le système. On peut émettre plusieurs hypothèses à propos du fonctionnement d'un système de files d'attente donné, c'est-à-dire à propos du mécanisme d'arrivée des clients, du mécanisme de service et de la discipline de gestion de la file d'attente.



**Fig. 2.1**– Structure de base d'une file d'attente. (Kobayashi 1978)

### 2.5.1 Schéma d'arrivée

Un client fait partie d'une population de clients. Une population est caractérisée par sa taille c'est-à-dire le nombre de clients potentiels. La taille d'une population peut être finie ou infinie. On doit faire explicitement l'hypothèse d'une population finie si le flux d'arrivée est affecté par le nombre de clients qui sont dans le système. On parlera d'un système ouvert si le nombre de clients est infini. Si, par contre, le nombre de client est fini, alors on parlera dans ce cas d'un système fermé.

Une autre caractéristique du schéma d'arrivée des clients est le processus d'arrivée dans le temps. Le processus le plus simple est celui de l'arrivée à taux constant; les clients arrivent à intervalles de temps réguliers de longueur  $\tau$ . Dans ce cas, le taux d'arrivée est  $\lambda = 1/\tau$  par unité de temps.

Le processus d'arrivée le plus simple et le plus utilisé en pratique est le processus d'arrivée aléatoire. On utilise en général le processus d'arrivée de Poisson. Nous traiterons de ce processus au paragraphe 2.6.

### 2.5.2 Le processus de service

Le second composant d'un système de file d'attente est le temps nécessaire pour servir un client pris individuellement.

L'unité de service varie en fonction de la nature du serveur et de ses clients. Par exemple, si le serveur est un CPU et les clients sont des programmes, une unité appropriée serait l'instruction. Si le serveur est un canal de transmission, l'unité de service serait le bit. Dans le cas du trafic aérien, l'unité de service s'avère l'avion. Dans la grande majorité des cas, on suppose que la population de clients est homogène, ce qui veut dire que les demandes de service ont une distribution commune dénommée "distribution de service". Dans certains cas plus complexes, les clients peuvent être classés selon différents types, chaque type de clients



possédant sa propre distribution de demande de service. La présence de différents types d'avions entraîne l'existence de différents temps de service.

On doit également spécifier la capacité du serveur c'est-à-dire la vitesse à laquelle le service demandé est effectué. Si la demande de service d'un client est  $S$  [unités de service], et le serveur correspondant a une capacité  $C$  [unités de service par sec], alors le rapport  $S/C$  [sec] est le temps de service. Sa valeur moyenne  $\bar{S}/C$  [sec] est appelée "le temps moyen de service", et son inverse  $\mu = C/\bar{S}$  est dénommé "le taux de réalisation" ou "le taux de service".

### 2.5.3 Stratégie de gestion de la file

La stratégie de gestion de la file d'attente pose la question de savoir dans quel ordre les clients qui se trouvent sur la file d'attente sont servis. Le système d'ordonnancement le plus familier est celui du "premier arrivé, premier servi" ou en anglais "first-come first-served" (FCFS)<sup>6</sup>. Mais on pourrait également adopter une stratégie "last-come, first-served" (LCFS)<sup>7</sup> ou "dernier arrivé, premier servi". Dans ces deux cas de figure, les clients sont servis selon leur ordre d'arrivée.

Mais dans certaines circonstances, quelques clients sont plus importants que d'autres et méritent de ce fait, un traitement de faveur par rapport aux autres. Dans le cadre de la problématique du trafic aérien, on pourrait considérer qu'un atterrissage sera prioritaire sur un décollage en ce qui concerne l'utilisation de la piste. Une procédure qui permet la différenciation parmi les clients est appelée "stratégie avec priorité". La règle de priorité détermine l'ordre de passage des clients qui attendent sur la file d'attente.

Pour la stratégie de gestion de la file, il existe deux types de priorité: la priorité préemptive et la priorité non préemptive.

Dans le cas de priorité préemptive, si un client est en train d'être servi, l'arrivée d'un client prioritaire entraîne le retour du client en cours de service dans la file d'attente. Par contre, dans un système à priorité non préemptive, l'arrivée d'un client prioritaire n'entraîne pas l'éjection du client en cours de service. Le client prioritaire qui vient d'arriver se place en tête de la file et attend la fin de service du client en cours.

<sup>6</sup> parfois appelé "first-in, first-out" (FIFO)

<sup>7</sup> parfois appelé "last-in first-out" (LIFO)

## 2.6 Processus de Poisson et ses propriétés

Dans la section 2.5.1, nous avons introduit le terme de processus de Poisson pour désigner un processus d'arrivée complètement aléatoire. Dans cette section, nous présentons une définition formelle du processus de Poisson et ses propriétés importantes.

Formellement un processus de Poisson est défini par cinq conditions ou postulats. En effet, si on considère un intervalle de temps fini  $(0, T)$ ,  $N(T)$  le nombre de survenances dans cet intervalle de temps,  $N(T)$  est un processus de Poisson s'il vérifie les conditions suivantes:

$$1- \forall T, N(T) \in \mathbb{N}$$

$$2- T_1 \cap T_2 = \emptyset \Rightarrow N(T_1 \cup T_2) = N(T_1) + N(T_2)$$

$$3- T_1 \cap T_2 = \emptyset \Rightarrow N(T_1) \text{ et } N(T_2) \text{ indépendants}$$

$$4- \Pr[N(T) = k] = P_k(|T|) \text{ où } |T| \text{ est la longueur de l'intervalle } T.$$

La probabilité d'avoir  $k$  survenances dans un intervalle ne dépend que de la longueur de ce dernier.

$$5- P_0(T) = 1 - \lambda T + O(T)$$

$$P_1(T) = \lambda T + O(T)$$

$$\text{Avec } \lim_{T \rightarrow 0} \frac{O(T)}{T} = 0$$

$\lambda$  représente le nombre de survenances pendant une tranche de temps unitaire.

Les deux conditions du postulat 5 ont pour conséquence qu'il ne peut y avoir de survenance multiple:

$$\begin{aligned} \sum_{k=2}^{\infty} P_k(T) &= 1 - P_0(T) - P_1(T) \\ &= 1 - (1 - \lambda T + \lambda T + O(T)) \end{aligned}$$



$$= O(T)$$

À partir de ces postulats, on peut déduire un certain nombre de propriétés.

### Distribution de $N(T)$

On peut démontrer que le nombre de survenances pendant l'intervalle de temps  $T$ ,  $N(T)$  est une variable aléatoire de Poisson de moyenne  $\lambda T$ .

$$\Pr[N(T) = k] = \frac{(\lambda T)^k}{k!} e^{-\lambda T}$$

### Distribution du temps entre survenances

On peut démontrer que le temps entre survenances de Poisson a une distribution exponentielle de taux  $\lambda$ . En effet si  $Z$  désigne le temps entre survenances, nous avons:

$$\Pr[Z \leq a] = 1 - e^{-\lambda a}$$

### Superposition de processus de Poisson

Si on superpose plusieurs processus de Poisson indépendants, le processus résultant est aussi un processus de Poisson. On peut par exemple fusionner plusieurs processus d'arrivée à une file d'attente. Le processus d'arrivée résultant est un processus de Poisson si les processus initiaux le sont.

### Décomposition d'un processus de Poisson

Soit un processus de Poisson de taux  $\lambda$ . Ce processus d'arrivée se distribue en  $p$  processus sous-jacents suivant des proportions constantes  $q_1, \dots, q_p$ . Ceci correspond au tirage au sort des arrivées successives comme un tirage avec remise de boules de  $p$  couleurs dans une urne.

## 2.7 Distribution de service exponentielle

Nous n'allons pas dans cette partie présenter toutes les distributions de service qui existent, mais celle qui sera utilisée dans les modèles de files d'attente proposés dans ce travail.

On dit qu'une variable aléatoire  $X$  satisfait à la loi exponentielle de paramètre  $\lambda$  ( $\lambda$  réel positif donné) lorsque sa fonction de densité de probabilité  $f$  est définie par:

$$f(x) = \lambda e^{-\lambda x}, \quad \forall x \geq 0.$$

$X$  est appelé variable exponentielle. On peut démontrer que  $E(X) = Var(X) = \frac{1}{\lambda}$

La fonction de répartition  $F$  est définie par les relations suivantes:

$$F(t) = \begin{cases} 1 - e^{-\lambda t} & \forall t \geq 0 \\ 0 & \forall t < 0 \end{cases}$$

## 2.8 Notations de Kendall

Pour spécifier une file d'attente donnée, on utilise la notation  $A/B/C/K/m/Z$ , où  $A$  et  $B$  désignent respectivement le processus d'arrivée et le processus de service.  $C$  représente le nombre de serveurs parallèles,  $K$  la capacité maximale du système,  $m$  la population des usagers et  $Z$  la discipline de service.

Dans la description des processus d'entrée ou des processus de service, la liste suivante donne les conventions fréquemment utilisées pour symboliser les distributions.

- $M$  Distribution exponentielle. Le phénomène de "sans mémoire" de la distribution exponentielle est aussi référencée comme la propriété markovienne.
- $D$  Processus fixe, déterministe. Le temps interarrivée et le temps de service sont des constantes.
- $Ek$  Distribution d'Erlang à  $k$  phases.
- $Hk$  Distribution hyperexponentielle à  $k$  phases.



*G* Distribution quelconque.

*GI* Distribution générale indépendante (en ce qui concerne le temps interarrivée).

Les principales disciplines de service utilisées sont:

FCFS ou FIFO: "premier arrivé, premier servi" (le terme anglais est "first-come, first served"),

LCFS ou LIFO: "dernier arrivé, premier servi" ("last-come, first-served" en anglais),

A ou FIRO: aléatoire ("first in random out").

Lorsque les trois derniers éléments de la notation de Kendall ne sont pas précisés, il est sous-entendu que  $Z = \text{FCFS}$ ,  $m = +\infty$  et  $K = +\infty$ .

## 2.9 Mesures de performance d'une file d'attente

La motivation sous-jacente pour la construction d'un modèle mathématique d'un système est de mesurer les performances de ce dernier. Si le système est une file d'attente, une mesure pertinente sera la congestion. Une mesure simple de la congestion est l'*Intensité du trafic*. Cette quantité adimensionnelle est définie de la manière suivante:

$$\text{Intensité du trafic} = \frac{\text{Temps de service moyen}}{\text{Temps moyen entre arrivées}}$$

Si dans une population infinie, le taux d'arrivée est  $\lambda$  et le temps de service moyen est  $1/\mu$  alors, on peut écrire:

$$\text{Intensité du trafic} = \text{Taux d'arrivée} * \text{Temps de service moyen} = \frac{\lambda}{\mu}$$

Une intensité de trafic supérieure à 1 signifie que les clients arrivent plus vite que le serveur ne les prend en charge. s'il y a  $m$  serveurs en parallèle, chaque serveur reçoit  $\lambda/m$  clients par unité de temps. Ainsi un  $m$ -serveur peut gérer une intensité de trafic jusqu'à  $m$ . Le

facteur d'utilisation que l'on note  $\rho$  représente la fraction de temps pendant laquelle un serveur est occupé.

Si nous avons  $m$  serveurs qui travaillent en parallèle, alors nous avons la relation:

$$\text{Facteur d'utilisation} = \rho = \min\left\{\frac{\lambda}{m\mu}, 1\right\}$$

Pour un seul serveur,  $\rho = \lambda/\mu$ . Ainsi le facteur d'utilisation et l'intensité du trafic sont égaux si  $\rho = \lambda/\mu < 1$ .

On peut également mesurer le nombre moyen de clients gérés par unité de temps – le throughput –. Dans un système avec  $m$  serveurs en parallèle, il y a  $m\rho\mu$  clients gérés par unité de temps. Ainsi nous avons:

$$\text{Throughput} = m\rho\mu = \min\{\lambda, m\mu\}$$

Pour le client, la mesure la plus importante est le temps passé à attendre sur la file d'attente ou le temps passé dans le système. On définit le temps d'attente du client (Waiting time)  $W_j$  comme étant le temps passé sur la file d'attente, tandis que le temps de réponse du système (Response time)  $R_j$  représente le temps total que le client  $j$  passe dans le système. Ainsi on peut écrire la relation suivante:

$$\text{Temps de réponse (R)} = \text{Temps d'attente (W)} + \text{Temps de service (S)}$$

$\{W_j; j = 1, 2, \dots\}$  et  $\{R_j; j = 1, 2, \dots\}$  sont des processus aléatoires. Pour la mesure de performance d'un système, on utilise généralement des valeurs moyennes  $\bar{R}$  et  $\bar{W}$ .

Une autre mesure de congestion intéressante est la taille de la file d'attente ou le nombre de clients présents sur la file d'attente.

Si  $Q(t)$  représente au temps  $t$  le nombre de clients qui attendent d'être servis et si  $N(t)$  représente le nombre total de clients dans le système au temps  $t$ , alors nous avons dans le cas d'un système avec  $m$  serveurs la relation suivante:  $Q(t) = \max\{0, N(t) - m\}$ .



## 2.10 Formule de Little: $L = \lambda W$

Probablement, la formule la plus importante et la plus simple utilisée dans l'étude des systèmes de file d'attente est la formule  $L = \lambda W$ , connue sous le nom de la formule de Little ou théorie de Little. Cette formule s'écrit  $\bar{Q} = \lambda \bar{W}$  en considérant les conventions d'écriture adoptées en amont.  $\bar{Q}$  représente la longueur moyenne de la file d'attente,  $\lambda$  le taux moyen d'arrivée et  $\bar{W}$  le temps d'attente moyen dans la file.

Cette formule peut encore s'écrire sous la forme  $\bar{N} = \lambda \bar{R}$  où  $\bar{N}$  représente le nombre moyen de clients dans le système et  $\bar{R}$  le temps de réponse moyen.

## 2.11 Architecture générale du modèle

Avant tout décollage, la compagnie aérienne introduit auprès du CMFU un plan de vol. Ce plan contient les différentes étapes du vol et les différentes régions à survoler. On considère alors un vol comme une succession de temps de services dispensés par des serveurs. L'avion est le "client". Les serveurs sont de différentes sortes. Certains serveurs ont une capacité limitée si bien qu'à cet endroit se forme une file d'attente qui occasionne un temps d'attente. Comme exemple, nous pouvons citer un aéroport qui dispose d'une seule piste d'atterrissage. Si plusieurs avions sollicitent en même temps un atterrissage, la tour de contrôle doit pouvoir ordonnancer ces atterrissages. Certains avions devront attendre et ceci contribuera à augmenter la durée du vol. Le temps d'attente sur une file est aléatoire.

Les temps de service sont également aléatoires, car ils dépendent des facteurs exogènes comme la force du vent, la pluviosité, la visibilité, etc. L'ensemble du modèle est un réseau de files d'attente comprenant des serveurs de nature différente.

Avant de faire une étude du réseau dans son ensemble, nous allons proposer un modèle pour chaque serveur du réseau.

Les différents serveurs à considérer sont: la porte (GATE), le taxiway (TWY), la piste servant de décollage et/ou d'atterrissage (RWY), l'espace aérien de gestion de l'aéroport (TMA), et les différents secteurs (SEC). Pour aller d'une localité d'origine à une localité d'arrivée, l'avion traverse différents secteurs contrôlés par des centres de contrôle de trafic aérien (Air Traffic Control ou ATC).

Enocontrol?  
↓

### 2.11.1 La porte (GATE)

Dans un aéroport, la porte permet d'embarquer ou de débarquer des passagers. Elle permet dans le modèle global de réseaux de files d'attente à un avion d'entrer ou de sortir du réseau. L'entrée dans le réseau par une porte correspond à une arrivée devant une porte pour embarquer des passagers. La sortie du réseau correspond par exemple au passage d'un avion de la porte à un hangar d'entretien après avoir débarqué des passagers.

La file d'attente devant les portes de l'aéroport est donc composée d'avions qui veulent débarquer des passagers et ceux qui veulent embarquer des passagers.

Chaque porte ne peut servir qu'à un seul avion à la fois. Mais un même vol peut être servi par différentes portes. On peut toutefois faire l'hypothèse préliminaire qu'un même vol n'est servi que par une seule porte.

On peut répertorier différents types de portes. Des portes pouvant accommoder des B737 ne pourront pas accommoder des gros porteurs à l'instar des B747. Par contre, des B737 pourront être assignés à des portes prévues pour les B747, même si une perte de capacité en découle.

La stratégie de service est "premier arrivé, premier servi" (FCFS). Le temps de service a une distribution exponentielle. Le processus d'arrivée à la porte sera considéré comme un processus de Poisson.

Le modèle pour ce nœud conformément à la notation de Kendall est donc de type M/M/m.

Nous supposons que la file d'attente devant les  $m$  portes est illimitée, c'est à dire le nombre d'avions sollicitant le service d'une porte est illimité.

Le processus de naissance et de mort nous permet d'écrire:

$$\lambda_k = \lambda \quad k = 0, 1, 2, \dots$$

$$\mu_k = \min\{k\mu, m\mu\}$$



$$= \begin{cases} k\mu & 0 \leq k \leq m \\ m\mu & m \leq k \end{cases}$$

La condition d'ergodicité est  $\lambda/m\mu < 1$

Pour calculer  $P_k$  – la probabilité d'avoir  $k$  avions dans le système – de l'équation (2.9), nous devons séparer la solution en deux parties, parce que la dépendance de  $\mu_k$  en fonction de  $k$  est répartie entre deux intervalles:  $k \leq m$   $k \geq m$ . Il vient:

$$P_k = \begin{cases} P_0 \frac{(m\rho)^k}{k!} & k \leq m \\ P_0 \frac{\rho^k m^m}{m!} & k \geq m \end{cases}$$

$$\text{où } \rho = \frac{\lambda}{m\mu} < 1$$

Nous obtenons la valeur de  $P_0$  à partir de l'équation (2.9) du processus de naissance et de mort (Voir paragraphe 2.4).

$$\begin{aligned} P_0 &= \left[ 1 + \sum_{k=1}^{m-1} \frac{(m\rho)^k}{k!} + \sum_{k=m}^{\infty} \frac{(m\rho)^k}{m!} \frac{1}{m^{k-m}} \right]^{-1} \\ &= \left[ \sum_{k=0}^{m-1} \frac{(m\rho)^k}{k!} \left( \frac{(m\rho)^m}{m!} \right) \left( \frac{1}{1-\rho} \right) \right]^{-1} \end{aligned}$$

La probabilité pour qu'un avion soit forcé de joindre la file d'attente parce que toutes les portes sont occupées est donnée par la relation suivante:

$$\begin{aligned} P[\text{Attente}] &= \sum_{k=m}^{\infty} P_k \\ &= \sum_{k=m}^{\infty} P_0 \frac{(m\rho)^k}{m!} \frac{1}{m^{k-m}} \end{aligned}$$

Il vient:

$$P[Attente] = \frac{\left( \frac{(m\rho)^m}{m!} \right) \left( \frac{1}{1-\rho} \right)}{\sum_{k=0}^{m-1} \frac{(m\rho)^k}{k!} + \left( \frac{(m\rho)^m}{m!} \right) \left( \frac{1}{1-\rho} \right)}$$

Les paramètres qui seraient intéressants de connaître pour la gestion des portes au niveau d'un aéroport sont:

$\bar{M}$  : nombre moyen de portes occupées. C'est aussi le nombre d'avions en cours de service;

$\bar{Q}$  : nombre moyen d'avions qui attendent sur la file qu'une porte se libère;

$\bar{W}$  : temps d'attente moyen dans la file;

$\bar{N}$  : nombre moyen d'avions dans le système,  $\bar{N} = \bar{Q} + \bar{M}$  ;

$\bar{S}$  : temps moyen de service ou temps moyen que passe un avion devant une porte;

$\bar{R}$  : temps de réponse moyen du système,  $\bar{R} = \bar{W} + \bar{S}$ .

$$\bar{M} = \sum_{k=1}^{m-1} kP_k + m \frac{P_m}{1-\rho}$$

$$= m\rho$$

$$\bar{N} = \sum_{k \geq 0} kP_k$$

$$= m\rho + \rho \frac{(m\rho)^m}{m!} \frac{P_0}{(1-\rho)^2}$$

Le nombre moyen d'avions dans le système est égal à la somme du nombre d'avions en service et du nombre d'avions en attente -  $\bar{N} = \bar{Q} + \bar{M}$  -. À partir de cette relation, on peut déduire  $\bar{Q}$ .

L'équation de Little nous permet de calculer  $\bar{R}$  et  $\bar{W}$ .

$$\bar{N} = \lambda \bar{R} \Leftrightarrow \bar{R} = \frac{1}{\lambda} \bar{N}$$

$$\bar{Q} = \lambda \bar{W} \Leftrightarrow \bar{W} = \frac{1}{\lambda} \bar{Q}$$



Le temps moyen de service s'obtient par la relation  $\bar{R} = \bar{W} + \bar{S}$ .

Il est intéressant d'avoir la fonction de distribution du temps moyen d'attente sur la file d'attente  $F_W(t)$ .

$$F_W(t) = \Pr[\bar{W} \leq t] = 1 - \frac{P_m}{1-\rho} + \frac{P_m}{1-\rho} (1 - e^{(\mu m - \lambda)t})$$

Ainsi, s'il y a attente, cette attente a une distribution exponentielle de taux  $\mu m - \lambda$

### 2.11.2 Le taxiway (TWY)

Pour aller de la porte d'embarquement à la piste de décollage ou pour aller de la piste d'atterrissage à la porte de débarquement, l'avion emprunte le taxiway.

Le même taxiway peut servir un grand nombre d'avions en même temps. La capacité est généralement considérée comme infinie, même si elle est parfois utilisée comme une aire d'attente.

La distribution du temps de trajet (ou temps de service) ne nous est pas actuellement connue. Nous considérons dans une première approche le TWY comme un serveur exponentiel sans attente (infinité de serveurs).

Nous pouvons dire que le taxiway est un modèle du type  $M/M/\infty$  d'après la notation de Kendall. Le processus d'arrivée est celui de Poisson et le processus de service est exponentiel. Le fait qu'il existe une infinité de serveurs implique l'absence de file d'attente devant le TWY. Cette propriété peut être interprétée comme un serveur qui réagit de manière linéaire pour empêcher la formation d'une file d'attente.

Nous avons:

$$\lambda_k = \lambda, k = 0, 1, 2, \dots$$

$$\mu_k = k\mu, k = 0, 1, 2, \dots$$

Nous obtenons à partir de l'équation (2.9) la solution suivante:

$$P_k = P_0 \prod_{i=0}^{k-1} \frac{\lambda}{(i+1)\mu}$$

$$= \frac{(\lambda/\mu)^k}{k!} e^{-\lambda/\mu} \quad k=0, 1, 2, \dots$$

Le nombre moyen d'avions dans le système TWY est  $\bar{N} = \frac{\lambda}{\mu}$

La condition d'ergodicité est  $\lambda/\mu < \infty$ .

L'équation de Little nous permet de calculer le temps de réponse moyen du système.

$$\bar{N} = \lambda \bar{R}$$

$$\bar{R} = \frac{1}{\lambda} \bar{N}$$

### 2.11.3 La piste (RWY)

Près de la moitié des aéroports les plus importants ne disposent que d'une seule piste qui fait office non seulement de piste d'atterrissage mais aussi de piste de décollage.

Pour la gestion d'une piste, deux modèles de file d'attente doivent être envisagés.

- Lorsque la piste ne sert qu'à l'atterrissage ou qu'au décollage, le serveur est de type classique, c'est-à-dire un seul serveur, une seule distribution de temps de service, une stratégie de la gestion de la file FCFS. Dans ce modèle, tous les avions sont servis de façon identique.
- Lorsque la piste est utilisée à la fois pour l'atterrissage (A) et pour décollage (D), il faut envisager deux classes de clients (A et D) avec priorité relative pour les clients de classe A sur les clients de classe D, c'est-à-dire qu'un avion ne peut décoller que s'il n'y a pas d'avions qui demandent à atterrir.

Considérons le premier cas où la piste ne sert qu'à l'atterrissage ou qu'au décollage. Dans cette situation, nous sommes en présence du modèle  $M/M/1$  avec une gestion de la file d'attente FCFS.



Le système de file d'attente le plus connu  $M/M/1$  est le plus simple et le moins trivial. Ce système peut être décrit en sélectionnant les coefficients du processus de naissance et de mort de la manière suivante:

$$\lambda_k = \lambda, k = 0, 1, 2, \dots$$

$$\mu_k = \mu, k = 0, 1, 2, \dots$$

Nous posons tous les coefficients de naissance égaux à la constante  $\lambda$  et tous les coefficients de mort égaux à  $\mu$ . Nous supposons en plus que la longueur de la file d'attente est infinie c'est-à-dire qu'on peut avoir un très grand nombre d'avions qui attendent pour atterrir ou pour décoller.

Si dans l'équation (2.9), nous appliquons ces coefficients, nous obtiendrons:

$$P_k = P_0 \prod_{i=0}^{k-1} \frac{\lambda}{\mu}$$

ou

$$P_k = P_0 \left( \frac{\lambda}{\mu} \right)^k \quad k \geq 0 \quad (2.10)$$

Les conditions pour que notre système soit ergodique (et pour que, de cette façon, il admette une solution à l'équilibre  $P_k > 0$ ) sont  $S_1 < \infty$  et  $S_2 = \infty$ . Dans ce cas, la première condition devient:

$$S_1 = \sum_{k=0}^{\infty} \frac{P_k}{P_0} = \sum_{k=0}^{\infty} \left( \frac{\lambda}{\mu} \right)^k < \infty$$

La série à gauche de l'inégalité converge, si et seulement si  $\lambda/\mu < 1$ . La seconde condition d'ergodicité devient:

$$S_2 = \sum_{k=0}^{\infty} \frac{1}{\lambda(P_k/P_0)} = \sum_{k=0}^{\infty} \frac{1}{\lambda} \left( \frac{\mu}{\lambda} \right)^k = \infty$$

Cette dernière condition sera satisfaite si  $\lambda/\mu \leq 1$ ; ce qui entraîne qu'une condition d'ergodicité nécessaire et suffisante dans le système  $M/M/1$  est simplement  $\lambda < \mu$ . Pour déduire  $P_0$ , on utilise l'équation (2.9') et on obtient:

$$P_0 = \left[ 1 + \sum_{k=1}^{\infty} \left( \frac{\lambda}{\mu} \right)^k \right]^{-1}$$

La somme converge pour  $\lambda < \mu$  et

$$P_0 = \left[ 1 + \frac{\lambda/\mu}{1 - \lambda/\mu} \right]^{-1}$$

Il vient que:

$$P_0 = 1 - \frac{\lambda}{\mu} \quad (2.11)$$

En posant  $\rho = \lambda/\mu$ , la condition de stabilité demande que  $0 \leq \rho < 1$ ; il faut noter que cela assure que  $P_0 > 0$ .

De l'équation (2.10), nous avons finalement:

$$P_k = (1 - \rho)\rho^k \quad k = 0, 1, 2, \dots \quad (2.12)$$

L'équation précédente (2.12) est, en effet, la probabilité stationnaire de trouver  $k$  avions dans le système.

Une variable importante dans l'étude de ce système de file d'attente est le nombre moyen d'avions dans le système  $\bar{N}$ . Ce qui est donné par:

$$\begin{aligned} \bar{N} &= \sum_{k=0}^{\infty} k P_k \\ &= (1 - \rho) \sum_{k=0}^{\infty} k \rho^k \\ &= \frac{\rho}{1 - \rho} \end{aligned} \quad (2.13)$$



En appliquant directement la relation de Little à l'équation (2.13) pour obtenir  $\bar{R}$ , le temps moyen que passe un avion dans le système est donné par:

$$\begin{aligned}\bar{R} &= \frac{\bar{N}}{\lambda} \\ \bar{R} &= \left( \frac{\rho}{1-\rho} \right) \left( \frac{1}{\lambda} \right) \\ \bar{R} &= \frac{1/\mu}{1-\rho}\end{aligned}\tag{2.14}$$

La valeur que prend  $\bar{R}$  quand  $\rho = 0$  est exactement le temps de service moyen escompté par un avion; c'est-à-dire qu'il n'attend pas sur la file d'attente et  $1/\mu$  seconde en service en moyenne.

L'analyse du comportement du système à partir des équations (2.13) et (2.14) suscite quelques remarques pour une bonne gestion de la piste. En effet, au fur et à mesure que la valeur de  $\rho$  approche de l'unité, c'est-à-dire lorsque le taux d'arrivées  $\lambda$  des avions avoisine le taux de service de la piste  $\mu$ , le nombre moyen d'avions dans le système et le temps de réponse moyen augmentent de façon significative. On peut observer une instabilité du système lorsque  $\rho=1$ , ce qui n'est pas surprenant lorsque la condition d'ergodicité est  $\rho < 1$ . Ce qui pourrait surprendre, c'est le nombre moyen d'avions dans le système  $\bar{N}$  et le temps de réponse du système  $\bar{R}$  qui se détériorent quand  $\rho \rightarrow 1$ . On peut dire qu'on risque de paralyser un aéroport, lorsqu'on fait fonctionner la piste en s'approchant de la valeur de sa capacité. Une explication intuitive serait qu'avec un flux d'avions aléatoire, nous avons des pics de trafic avec surcharge du système; tandis qu'il est vrai que la piste restera libre en moyenne  $1-\rho=P_0$  du temps. Le temps pendant lequel la piste restera libre n'aura pas une distribution uniforme dans les petits intervalles de temps contrairement au long terme.

À ce stade de l'analyse, nous pouvons présager que dans les hypothèses fixées, le serveur piste pourrait constituer un facteur limitant de la performance du système.

Un autre paramètre intéressant à calculer est la probabilité de trouver au moins  $k$  avions dans le système:

$$\begin{aligned}
 P[\geq k] &= \sum_{i=k}^{\infty} P_i \\
 &= \sum_{i=k}^{\infty} (1-\rho)\rho^i
 \end{aligned}$$

$$P[\geq k] = \rho^k$$

La probabilité de dépasser un certain nombre d'avions dans le système est une fonction qui décroît géométriquement avec ce nombre.

D'autres variables permettant d'évaluer les performances du système peuvent être obtenues aisément.

$$\bar{W} = \bar{R} - \bar{S}$$

$$\bar{W} = \frac{1}{\mu(1-\rho)} - \frac{1}{\mu} = \frac{\rho}{\mu(1-\rho)}$$

Si  $Q(t)$  désigne le nombre d'avions en attente sur la file à l'instant  $t$ , et  $\bar{Q}$  désigne sa valeur moyenne, alors en appliquant la formule de Little, on obtient:

$$\bar{Q} = \lambda \bar{W} = \frac{\rho^2}{1-\rho}$$

Notons que le nombre moyen d'avions trouvés sur la piste est:

$$\bar{N} - \bar{Q} = \rho$$

On peut démontrer que la fonction de distribution du temps d'attente moyen sur la file d'attente  $F_W(t)$  est la suivante:

$$F_W(t) = \Pr[\bar{W} \leq t]$$

$$F_W(t) = 1 - \rho e^{-t\mu(1-\rho)}$$



### 2.11.4 L'espace aérien de gestion de l'aéroport (TMA)

C'est une zone proche d'un aérodrome. Elle fait la transition entre la piste et le secteur. Juste avant l'atterrissage ainsi qu'après le décollage, les avions se trouvent dans le TMA.

Il existe peu de données sur le temps de service du TMA. C'est un système sans attente, mais avec une capacité limitée. Nous pouvons formuler l'hypothèse que le TMA est un système  $M/M/\infty$  semblable à celui du TMY. Les résultats obtenus lors de l'analyse du taxiway sont, de cette manière, transposables au TMA.

### 2.11.5 Le secteur (SEC)

Le secteur est un morceau d'espace aérien de taille variable. Il contient plusieurs segments de routes aériennes et est surveillé au minimum par deux contrôleurs. Même si le secteur peut servir plusieurs vols simultanément, sa capacité reste limitée.

Le temps de service d'un secteur peut être prolongé afin de résoudre les conflits qui pourraient subvenir entre avions. Le nombre d'avions dans le secteur et le nombre de contrôleurs en poste sont également des facteurs susceptibles de faire varier le temps de service d'un secteur.

Lorsqu'un avion entre dans un secteur, il est immédiatement pris en charge par un contrôleur qui analyse sa trajectoire et qui donne des instructions au moment où l'avion est en conflit potentiel avec un autre. Le rôle du contrôleur au sein du système est à ce point important que c'est le nombre de contrôleurs en poste qui détermine l'intensité du trafic dans le secteur. Si les contrôleurs ne sont pas assez nombreux, des problèmes peuvent se poser, même si le secteur fonctionne en dessous de sa capacité. C'est pourquoi nous assimilerons le serveur secteur à la population des contrôleurs en poste qui aiguillent les aéronefs.

En supposant qu'il y ait  $m$  contrôleurs disponibles à un poste de contrôle ATC pour un secteur donné, que le processus d'arrivée des avions dans le secteur soit complètement aléatoire (Poisson) et que le temps de service suive une distribution exponentielle, nous pouvons considérer le système comme un modèle  $M/M/m$  suivant la notation de Kendall. Dans ces conditions, les résultats obtenus pour l'étude du système de portes (Gates) sont transposables au serveur secteur.

Dans un modèle global, on pourrait considérer un secteur comme un serveur sans attente où le temps de service dépendrait du nombre de clients.

Établir des listes reprenant tous les composants et tous les paramètres susceptibles d'influencer les performances d'un système ne constitue pas une des tâches les plus complexes. En revanche, il est moins aisé d'identifier un ensemble de paramètres critiques, et il est encore plus difficile de trouver des relations ou des équations qui rendent compte, selon ces paramètres, des performances du système dans son entièreté. Les difficultés rencontrées sont dues principalement au fait que le système global représente plus que la somme de ses composants.

Les modèles que nous avons élaborés jusqu'à présent ont été étudiés séparément. Tous les serveurs analysés ont été ainsi considérés indépendamment de leur contexte de fonctionnement. Dans un système global qui intègre le vol dans sa totalité, différents serveurs travaillent de manière synchrone et concurrentielle. Un réseau de files d'attente nous permet d'avoir un cadre mathématique de travail pour les problèmes d'allocation des ressources. Dans le Chapitre 4, nous intégrerons toutes les files d'attente en un modèle unique afin de constituer un réseau de files d'attente.



## Chapitre 3

# Réseaux de files d'attente

### 3.1 Généralités

Un réseau de files d'attente est un ensemble de stations de service qui possèdent chacune une file d'attente propre. Nous pouvons décrire le mécanisme d'un réseau de files d'attente comme suit: après avoir été servis à une station de service, les clients intègrent une autre file d'attente afin d'obtenir auprès d'une autre station de service un nouveau service.

Deux types de réseaux de files d'attente sont à distinguer: les réseaux fermés et les réseaux ouverts. Si le réseau de files d'attente est composé de  $M$  stations de service qui sont séparées et étiquetées  $1, 2, \dots, M$  et qui possèdent chacune une file d'attente propre, nous pouvons considérer un réseau de files d'attente fermé (comme le montre la figure 3.1(a)) comme un graphe  $G = (V, E)$ , où  $V = \{1, 2, \dots, M\}$  est l'ensemble de nœuds représentant les  $M$  stations de service et où  $E = V \times V$  est l'ensemble des arcs qui relient les stations.<sup>8</sup> Nous définissons le processus multidimensionnel (ou vecteur) de la façon suivante:

$$\underline{N}(t) = [N_1(t), N_2(t), \dots, N_i(t), \dots, N_M(t)]$$

où  $N_i(t)$  représente le nombre de clients (en attente ou en service) à la station  $i$  à l'instant  $t$ . Dans un réseau de files d'attente fermé, il n'y a pas d'arrivées de l'extérieur ou de départ vers l'extérieur; ce qui a pour conséquence que le nombre total de clients dans la système est constant.

$$N = \|\underline{N}(t)\| = \sum_{i=1}^M N_i(t)$$

<sup>8</sup>  $E = V \times V$  (le produit cartésien de  $V$  avec lui-même) est l'ensemble de toutes les paires ordonnées de la forme  $\langle i, j \rangle$  où  $i \in V$  et  $j \in V$ . L'élément  $\langle i, j \rangle$  représente le chemin de la station  $i$  à la station  $j$ .

En revanche, un réseau de files d'attente ouvert comporte une "source", une population infinie et une "destination", laquelle absorbe tous les clients qui quittent le réseau, comme illustré à la figure 3.1(b). Nous faisons l'ensemble des hypothèses suivantes:

- 1- Le flux de clients généré au nœud source est un processus de Poisson de taux  $\lambda(N)$  avec  $\|N(t)\| = N$ . Un client qui arrive se rend d'abord à la station  $i$  avec une probabilité  $q_{si}$ ,  $1 \leq i \leq M$ . Les propriétés du processus de Poisson nous permettent de dire que les sous-flux générés à chaque sous-station sont des processus de Poisson indépendants avec un taux  $\lambda(N)q_{si}$ ,  $1 \leq i \leq M$ .
- 2- La demande de service d'un client à une station  $i$  est une distribution exponentielle de moyenne  $\bar{S}_i$ ,  $1 \leq i \leq M$ . La capacité du processus de service de la station  $i$  peut dépendre de la taille de la file d'attente  $n_i = N_i(t)$  dans cette station à un temps donné. Nous notons sa capacité fonctionnelle par  $C_i(n_i)$ ,  $1 \leq i \leq M$ . De cette manière, le taux de réalisation à la station  $i$  est donné par la relation:
 
$$\mu_i(n_i) = \frac{C_i(n_i)}{\bar{S}_i}$$
- 3- La gestion de la file d'attente est quelconque.
- 4- Le routage d'un client est gouverné par une chaîne de Markov. En effet, la probabilité qu'un client parte de la station  $i$  pour la station  $j$  est donnée par  $q_{ij}$ ,  $j = 1, 2, \dots, M$  et  $d$  (destination). Nous supposons que la chaîne de Markov est irréductible.

De cette façon, le vecteur  $N(t)$  peut être considéré comme un processus multidimensionnel de naissance et de mort. Nous pouvons définir  $N(t)$  comme l'état du réseau de files d'attente au temps  $t$ .

Dans de nombreux problèmes d'un intérêt pratique, un modèle avec une population infinie de clients est inadéquat ou irréaliste. C'est pourquoi fréquemment, on doit recourir aux réseaux de files d'attente fermés au sein desquels le nombre total de clients  $N = \|n\|$  reste constant.



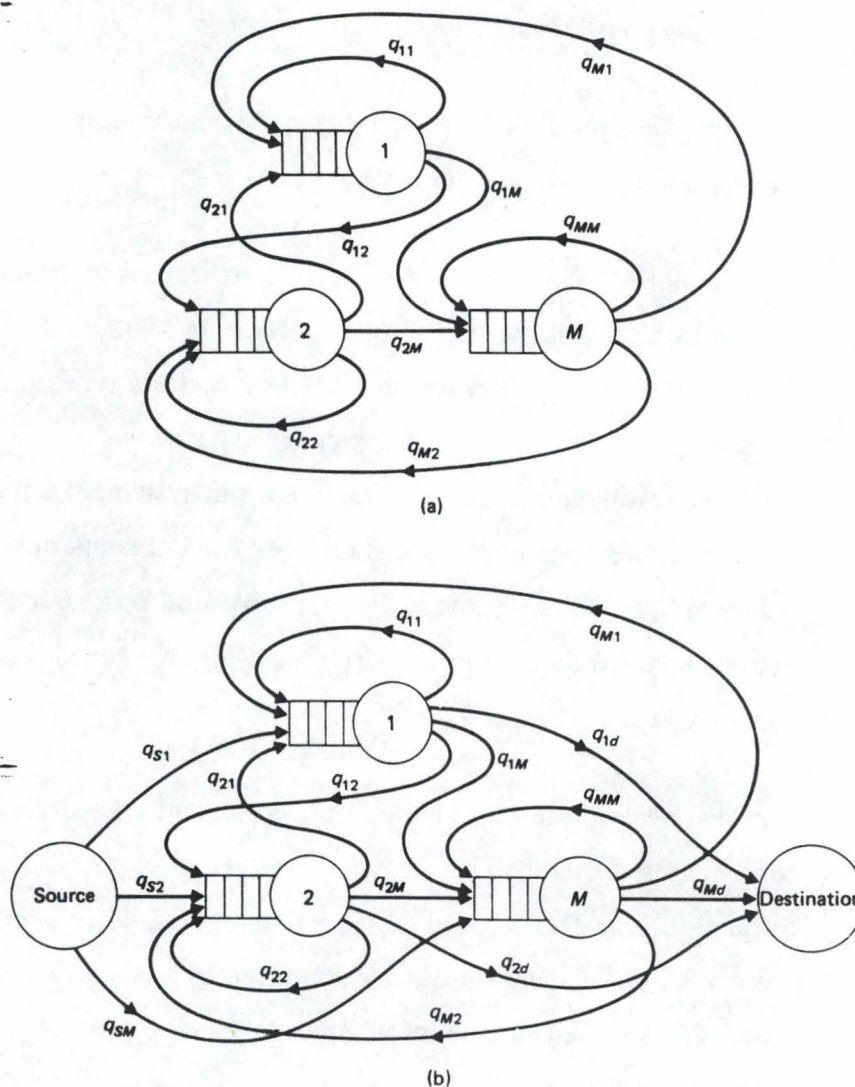


Figure 3.1– (a) Un réseau de files d'attente fermé et (b) un réseau ouvert (Kobayashi 1978)

### 3.2 Réseau de files d'attente pour la navigation aérienne

Le réseau est un ensemble de serveurs tels qu'ils ont été décrits et analysés individuellement au chapitre 2. Les serveurs de ce réseau sont les suivants:

- la porte (GATE) qui permet aux avions d'embarquer ou de débarquer des passagers;
- le taxiway (TMY) qui relie la piste au système de Gates;
- la piste (RWY) qui est utilisée pour le décollage et l'atterrissage des avions;

- l'espace aérien de gestion de l'aéroport (TMA) qui est l'espace aérien qu'emprunte l'avion peu après le décollage ou peu avant l'atterrissage;
- le secteur qui est un des morceaux d'espace aérien que l'on trouve après le TMA. Un avion en vol de croisière doit traverser différents secteurs.

Le secteur et la porte constituent les deux aires du réseau où les avions peuvent entrer dans le système ou sortir du système. En ce qui concerne le secteur, si nous analysons un réseau à l'échelle européenne, nous devons tenir compte de trois types de vols qui entrent et/ou sortent du système par les secteurs. Le premier type de vol qui est en provenance d'autres continents et à destination de l'Europe entre dans le système. Le second type de vol qui est en provenance de l'Europe et à destination d'autres continents quitte le système. Le troisième type de vol qui est en provenance d'autres continents à destination d'autres continents est en transit dans l'espace européen. Tous ces vols sont sous le contrôle de l'espace aérien européen au niveau des secteurs.

Pour ce qui est de la porte (GATE), elle est également considérée comme un point d'entrée ou de sortie du réseau, car il existe un espace réservé à l'entretien des avions. En effet, après avoir débarqué les passagers et avant de reprendre les airs, les avions doivent subir des opérations d'entretien qui s'effectuent sur des espaces réservés ou dans des hangars. Ils sortent ainsi du réseau par les Gates. De la même façon, après le passage dans un hangar, c'est par les Gates que les avions rentrent dans le réseau.



## Chapitre 4

# L'analyse de la stabilité, de l'optimisation et de la performance des réseaux de files d'attente multiclassées de Markov par une approche de programmation linéaire

L'objet de ce chapitre est d'étudier les réseaux de files d'attente comportant plusieurs classes d'avions. Dans la littérature, ces modèles sont appelés *Multiclass Queueing Networks*, dénomination abrégée en *MQNET*. Ces réseaux comprennent un ensemble de stations de services interconnectées. Dans ce modèle, le schéma d'arrivée des clients venant de l'extérieur est un flux indépendant de *Poisson*. De plus, les temps de service au niveau des stations sont exponentiels et le routage des clients entre les stations est markovien.

De récents résultats basés sur l'approche de *Programmation Linéaire* (LP) permettent d'établir une condition de stabilité simple et intuitive, ce qui rend possible une étude plus approfondie des questions d'ordonnancement optimal et une analyse des performances. La méthodologie exposée dans ce chapitre se base également sur la méthode de *programmation linéaire*. L'approche "primal-dual" repose sur le fait que le système satisfait aux lois de conservation afin de fournir des garanties de performance pour une heuristique d'ordonnancement basée sur un index. En outre, il sera possible d'analyser la performance d'un système de priorité arbitraire.

Les réseaux de files d'attente multiclassées (*MQNETs*) fournissent un champ riche de modèles pour une série de systèmes de service complexes comme, par exemple, pour le système de communication des ordinateurs (Gelenbe & Mitrani 1980). Le besoin d'évaluer et d'améliorer la performance de tels systèmes a motivé de nombreuses recherches ayant pour objet l'analyse, l'optimisation et la stabilité des *MQNETs*. Le modèle que nous présenterons dans ce chapitre est un *MQNET* "multistations" comprenant des stations à serveurs uniques. Il sera décrit dans la deuxième partie de ce chapitre. La troisième partie présentera, quant à elle,



l'étude de la stabilité. Dans la quatrième partie, il sera question de l'optimisation approximative reposant sur l'ordonnancement des clients, lui-même basé sur un index. Enfin, l'analyse de la performance clôturera ce chapitre.

La recherche de stabilité s'avère un préliminaire indispensable pour résoudre les problèmes d'optimisation et de performance. C'est un domaine très complexe où s'est affirmé ces dernières années un grand effort de recherche. Bon nombre de chercheurs ont focalisé leur attention sur le développement des tests computationnels afin d'obtenir une plus grande stabilité du système via sa performance dans un cadre non oisif en programmation linéaire (Kumar & Meyn 1996). La condition de stabilité simple et intuitive des *MQNETs* que nous présentons peut être formulée de la manière suivante: "si chaque station a une capacité de service assez grande pour gérer ses pics d'intensité de trafic, alors le réseau est stable sous la classe de discipline de service non oisive stationnaire". Bien que cette condition ne permet pas les meilleurs résultats, elle semble convenir pour résoudre les problèmes d'évaluation de performance et d'optimisation.

De récentes recherches réalisées par Papadimitriou et Tsitsiklis (1994) suggèrent que le problème d'optimisation de la performance, à savoir concevoir une règle d'ordonnancement qui optimise un objectif donné et qui calcule le système de performance correspondant, s'avère, dans la majorité des modèles *MQNET*, imprévisible sur le plan computationnel. Par conséquent, les chercheurs se sont efforcés de concevoir des règles d'ordonnancement heuristiques dotées d'une bonne performance empirique ou asymptotique (Harrison & Wein 1989). Cependant, la performance de tels règles est rarement garantie, ce qui semble en contraste avec la situation d'ordonnancement déterministe, puisque cette dernière comprend la méthode "primal-dual" qui a permis de concevoir des algorithmes heuristiques garantissant la performance. Dans la méthode "primal-dual", on cherche à apporter simultanément une solution heuristique et une solution pour le "dual" (Goeman & Williamson 1996). La quatrième partie de ce chapitre présente une approche "primal-dual" analogue à celle évoquée ci-dessus afin de développer, pour le *MQNET* que nous avons choisi de décrire, une classe de règles d'ordonnancement index-priorité dotées de garanties de performance. Ces résultats découlent des travaux de Bertsimas et Niño Mora (1996a) et de Glazenbrook et Garbe (1997), au sein desquels la notion de "lois de conservation" (approximatives) joue un rôle primordial. Grâce au cadre de travail mis en place dans les parties précédentes, nous serons capables, à la fin de ce chapitre, d'analyser les performances d'une règle de priorité quelconque.



#### 4.1. Modèle *MQNET*

Considérons un réseau de files d'attente ouvert, markovien, et comportant  $N$  classes d'avions ainsi que  $M$  stations de service. Une station  $m \in \mathcal{M} = \{1, \dots, M\}$  dessert un ensemble délimité de classes d'avions  $C_m \subseteq \mathcal{N} = \{1, \dots, N\}$ , où  $C_1, \dots, C_M$  constituent une partition de  $\mathcal{N}$ . Le processus d'arrivée, dans le réseau, des avions de classe  $i$  (*i*-avions) est un processus de Poisson de taux  $\alpha_i$ . Ces avions de classe  $i$  (*i*-avions) sont servis à la station  $s(i)$  pendant un temps exponentiel de taux de service  $\mu_i$ . Après le service, un *i*-avion peut soit être routé vers une autre station de service comme un *j*-aéronef avec une probabilité  $p_{ij}$ , soit il peut quitter le réseau avec une probabilité  $p_{io} = 1 - \sum_{j \in \mathcal{N}} p_{ij}$ . Pour qu'un avion qui entre dans le réseau le quitte avec une probabilité égale à 1, il faut que la matrice  $\mathbf{I} - \mathbf{P}$  soit inversible. En outre, nous supposons que les processus d'arrivée, de service et de routage des aéronefs soient mutuellement indépendants.

Nous définissons d'autres paramètres importants pour l'analyse du *MQNET*. Le *taux d'arrivée total des j-avions* noté  $\lambda_j$  est donné par la solution de l'équation

$$\lambda_j = \alpha_j + \sum_{i \in \mathcal{N}} \lambda_i p_{ij}, \text{ pour } j \in \mathcal{N}.$$

L'*intensité du trafic nominal des j-avions* noté  $\rho_j$  est donné par la relation

$$\rho_j = \frac{\lambda_j}{\mu_j}, \text{ pour } j \in \mathcal{N},$$

et représente le taux par lequel les demandes de service des *j*-avions (internes et externes) sont enregistrées dans le réseau. On peut également utiliser le terme de "taux d'utilisation des serveurs". Étant donné un sous-ensemble de classes d'aéronefs  $S \subseteq \mathcal{N}$ , nous définissons de la même façon l'*intensité du trafic pour les S-avions* noté  $\rho(S)$  par

$$\rho(S) = \sum_{j \in S} \rho_j.$$

La *S-Charge de travail moyenne d'un j-avion* notée  $V_j^S$  représente le temps moyen résiduel de service dont un *j*-avions a besoin avant de quitter  $S$  le premier;  $j \in S$  et  $S \subseteq \mathcal{N}$ . Les  $V_j^S$  peuvent être calculées par la résolution du système d'équations linéaires

$$V_i^S = \frac{1}{\mu_i} + \sum_{j \in S} p_{ij} V_j^S, \text{ pour } i \in S, S \subseteq \mathcal{N} \quad (4.1)$$

Nous définissons également par extension les paramètres  $V_i^S$  pour  $i \in S^C = \mathcal{N} \setminus S$ , à partir de la solution de l'équation (4.1). Nous pouvons maintenant procéder à la définition de l'intensité du trafic externe pour les  $j$ -avions, noté  $\rho^o(S)$ ,

$$\rho^o(S) = \sum_{j \in S} \alpha_j V_j^S \quad (4.2)$$

Pour examiner les problèmes de stabilité dans le cadre de notre modèle *MQNET*, les notions conventionnelles d'intensité de trafic doivent être remplacées par les concepts de taux de pic du trafic. Le *pic d'intensité du trafic de la classe  $i$  dans la station  $m$* , noté  $R(i, m)$  est le taux maximum par lequel la demande de service des  $i$ -avions peut être satisfaite dans cette station, c'est-à-dire,

$$R(i, m) = \mu_i \sum_{j \in C_m} p_{ij} V_j^{C_m}, \text{ pour } i \in \mathcal{N}, m \in \mathcal{M}. \quad (4.3)$$

Le *pic d'intensité du trafic de la station  $m'$  vers la station  $m$* , noté  $\bar{R}(m', m)$ , est le taux maximum par lequel les avions peuvent passer de  $m'$  vers  $m$ , c'est-à-dire,

$$\bar{R}(m', m) = \max_{i \in C_{m'}} R(i, m), \text{ pour } m, m' \in \mathcal{M}, \text{ avec } m' \neq m. \quad (4.4)$$

Enfin, nous définissons le *pic d'intensité du trafic pour la station  $m$* , noté  $\bar{\rho}(m)$ , comme le taux maximum par lequel les avions peuvent être transférés dans cette station, c'est-à-dire,

$$\bar{\rho}(m) = \sum_{j \in C_m} \alpha_j V_j^{C_m} + \sum_{m' \in \mathcal{M} \setminus \{m\}} \bar{R}(m', m), \text{ pour } m \in \mathcal{M}. \quad (4.5)$$

L'évolution du réseau est gouvernée par une règle d'ordonnancement qui spécifie dynamiquement comment les serveurs sont alloués aux avions. Nous considérons que ces règles sont *non anticipatives* - c'est-à-dire qu'elles ne sont pas basées sur des informations futures -, *stationnaires* et *non oisives* - c'est-à-dire qu'un serveur ne peut pas être libre quand il y a des avions qui sollicitent le service-. De telles règles d'ordonnancement sont dites "admissibles".

L'état du système à l'instant  $t$  est donné par les variables aléatoires suivantes:

- $L_j(t)$ : nombre total des  $j$ -avions dans le système à l'instant  $t$ ;



- $B_j(t)$ : 1 si un  $j$ -avion est en service au temps  $t$ ; 0 sinon;
- $B^m(t)$ : 1 si le serveur  $m$  est occupé à l'instant  $t$ ; 0 sinon.

Notre analyse utilise fréquemment les mesures de performance suivantes, définies pour chaque règle admissible:

$$x_j = E[L_j] \quad \text{pour } j \in \mathcal{A}, \quad (4.6)$$

$$x_{ij} = E[L_j | B_i = 1] \quad \text{pour } i, j \in \mathcal{A}, \quad (4.7)$$

et

$$x_j^{0m} = E[L_j | B^m = 0] \quad \text{pour } j \in \mathcal{A}, m \in \mathcal{M} \quad (4.8)$$

où les espérances ci-dessus représentent les temps moyens. Une règle est dite "stable", si le temps moyen des avions dans le réseau est fini. Le résultat-clé suivant a été obtenu grâce au travail de Bertsimas et de Niño Mora (1996).

Nous utiliserons les conventions d'écriture suivantes:  $\mathbf{x} = (x_j)_{j \in \mathcal{A}}$ ,  $\mathbf{X} = (x_{ij})_{i,j \in \mathcal{A}}$ ,  $\mathbf{X}^0 = (x_j^{0m})_{j \in \mathcal{A}, m \in \mathcal{M}}$ ,  $\alpha = (\alpha_j)_{j \in \mathcal{A}}$  et  $\Lambda = \text{Diag}(\lambda)$  où  $\lambda = (\lambda_j)_{j \in \mathcal{A}}$ . Nous aurons également besoin de  $\mu(S) = \{m; S \cap C_m \neq \emptyset\}$  et  $|\mathcal{M}(S)|$ .

**Théorème 1** *Sous n'importe quelle règle d'ordonnancement stable et admissible, les mesures de performance  $\mathbf{x}$ ,  $\mathbf{X}$  et  $\mathbf{X}^0$  satisfont aux contraintes suivantes (Bertsimas & Niño Mora 1996):*

(a) *Contraintes d'état*

$$x_j = \sum_{i \in C_m} \rho_i x_{ij} + \{1 - \rho(C_m)\} x_j^{0m}, \quad \text{pour } j \in \mathcal{A}, m \in \mathcal{M}. \quad (4.9)$$

(b) *Contraintes de conservation du flux*

$$-\alpha \mathbf{x}' - \mathbf{x} \alpha' + (\mathbf{I} - \mathbf{P})' \Lambda \mathbf{X} + \mathbf{X}' \Lambda (\mathbf{I} - \mathbf{P}) = (\mathbf{I} - \mathbf{P})' \Lambda + \Lambda (\mathbf{I} - \mathbf{P}). \quad (4.10)$$

(c) *Contraintes de décomposition du travail*

$$\begin{aligned} \{|\mathcal{M}(S) - \rho^0(S)|\} \sum_{j \in S} V_j^S x_j &= \sum_{j \in S} \rho_j V_j^S + \sum_{i \in S^c \cap (\cup_{m \in \mu(S)} C_m)} \sum_{j \in S} \rho_i V_j^S x_{ij} \\ &+ \sum_{i \in S^c} \sum_{j \in S} (\lambda_i V_i^S - \rho_i) V_j^S x_{ij} \\ &+ \sum_{m \in \mu(S)} \sum_{j \in S} \{1 - \rho(C_m)\} V_j^S x_j^{0m}, \quad \text{pour } S \subseteq \mathcal{A} \end{aligned} \quad (4.11)$$

L'équation (4.9) est élémentaire et résulte de la considération des états possibles de la station  $m$  dans une situation d'équilibre. La contrainte (4.10) a été obtenue par Bertsimas, Pascalidis et Tsitsiklis (1994) et par Kumar et Kumar (1994). Elle pourrait être déduite de l'application de la loi de la conservation du flux  $L^- = L^+$  de la théorie des files d'attente (découverte par Burke (1956) et Finch (1959)) pour tous les sous-systèmes de *MQNET*, qui comporteraient soit une seule classe de clients soit des paires de classes. Le résultat de la décomposition du travail (4.11) est dû au travail de Bertsimas et Niño Mora (1996). Elle est déduite de (4.9) et (4.10) par des manipulations algébriques.

## 4.2 Stabilité du modèle MQNET

Nous présenterons dans cette partie une condition de stabilité simple et intuitive des *MQNETs* markoviens généraux. Cette condition peut être formulée de la façon suivante: si chaque station a une capacité de service suffisante pour faire face aux pics d'intensité de trafic, alors le réseau est stable sous une règle admissible quelconque. Bien que le résultat obtenu ne soit pas le meilleur, il est pertinent, proche de la perfection sous certaines conditions données et surtout, il paraît un outil intéressant pour traiter analytiquement des problèmes subséquents d'évaluation de performance et d'optimisation (voir paragraphes 4 et 5). De plus, cette condition implique des facilités quant au calcul de la borne supérieure du nombre moyen de clients dans le réseau (et, par extension, quant au calcul du délai d'attente moyen du client via le théorème de Little). Cette borne supérieure est toujours valide, quelle que soit la règle admissible, ce qui est démontré par le Théorème 1 et par de récents résultats concernant le rapport entre la stabilité et la performance via LP (Kumar & Meyn 1996).

Nous exposons ci-dessous les principaux résultats.

**Théorème 2** (Condition globale de stabilité et borne de performance) *Si la condition*

$$\rho(C_m) < 1 \text{ et } \bar{\rho}(m) < 1 \text{ pour } m \in \mathcal{M} \quad (4.12)$$

*est satisfaite, alors*

- (i) *le réseau est stable sous n'importe quelle règle admissible, et la chaîne de Markov qui représente son évolution a un moment exponentiel qui converge géométriquement;*



- (ii) sous n'importe quelle règle admissible, le nombre moyen d'avions dans le réseau à l'instant  $t$  est borné comme suit:

$$\sum_{j \in \mathcal{N}} E[L_j] \leq \sum_{m \in \mathcal{M}} \sum_{j \in C_m} \frac{\rho_j V_j^{C_m}}{(1 - \bar{\rho}(m)) V^{C_m}}, \quad (4.13)$$

où

$$V^{C_m} = \min_{j \in C_m} V_j^{C_m}$$

### Remarques:

1. On peut démontrer facilement que si  $\rho(C_m) < 1$ , alors  $\rho(C_m) < \bar{\rho}(m)$  et que la condition  $\bar{\rho}(m) < 1$  est au moins aussi forte que  $\rho(C_m) < 1$ . En outre, si  $\bar{\rho}(m) - \rho(C_m) \rightarrow 0$  quand  $\max_{i \in \mathcal{N} - C_m, j \in C_m} P_{ij} \rightarrow 0$  et si  $\rho(C_m) < 1$  est nécessaire pour la stabilité, alors cette condition (4.12) est proche de la réalité quand le flow de clients est léger.
2. Il faut noter à propos du Théorème 2 (i) que la condition (4.12) implique une très forte stabilité.

Comme nous l'avons souligné plus haut, la condition LP de stabilité de Kumar et Meyn (1996) joue un rôle primordial dans la démonstration du Théorème 2 énoncé ci-dessus.

### La condition de stabilité de la LP

La condition de stabilité de Kumar et Meyn (1996) s'appuie sur une performance de la LP non oisive (performance considérée comme possible, si  $\rho(C_m) < 1$  pour  $m \in \mathcal{M}$ ) introduite par Bertsimas, Paschalidis et Tsitsiklis (1994) et par Kumar et Kumar (1994):

$$Z = \max x_1 + \dots + x_N$$

soumis à

$$-\alpha \mathbf{x}' - \mathbf{x} \alpha' + (\mathbf{I} - \mathbf{P})' \Lambda \mathbf{X} + \mathbf{X}' \Lambda (\mathbf{I} - \mathbf{P}) = (\mathbf{I} - \mathbf{P})' \Lambda + \Lambda (\mathbf{I} - \mathbf{P}). \quad (4.14)$$

$$x_j = \sum_{i \in C_m} \rho_i x_{ij} = 0, \text{ pour } j \in \mathcal{N} \text{ et } m = s(j) \quad (4.15)$$

$$x_j = \sum_{i \in C_m} \rho_i x_{ij} \geq 0, \text{ pour } j \in \mathcal{N} \text{ et } m \in \mathcal{M} \setminus \{s(j)\} \quad (4.16)$$

$$\mathbf{x}, \mathbf{X} \geq 0. \quad (4.17)$$

Dans la performance LP, les variables  $\mathbf{x} = (x_j)_{j \in \mathcal{N}}$  et  $\mathbf{X} = (x_{ij})_{i,j \in \mathcal{N}}$  correspondent à la mesure de performance donnée par les formules (4.6) et (4.7).

**Théorème 3** *Si la performance LP non oisive est bornée, alors le réseau est stable sous une règle stationnaire non oisive quelconque. En outre, la chaîne de Markov qui représente son évolution a un moment exponentiel qui converge géométriquement.* (Kumar & Meyn 1996)

Nous pouvons maintenant procéder à la démonstration du Théorème 2. Pour ce faire, nous allons montrer que (4.12) implique que la performance LP non oisive soit bornée.

Supposons que  $\mathbf{x}$  et  $\mathbf{X}$  soient les solutions possibles de cette LP. Les éléments de la matrice  $\mathbf{X}^0 = (x_j^{0m})_{j \in \mathcal{N}, m \in \mathcal{M}}$  sont donnés par

$$x_j^{0m} = \frac{1}{1 - \rho(C_m)} (x_j - \sum_{i \in C_m} \rho_i x_{ij}) \quad (4.18)$$

Il faut noter qu'à partir de (4.15) et de (4.16),

$$x_j^{0m} \geq 0 \text{ pour } j \in \mathcal{N}, m \in \mathcal{M}$$

et

$$x_j^{0m} = 0 \text{ pour } s(j) = m. \quad (4.19)$$

Par les commentaires qui suivent le Théorème 1,  $\mathbf{x}$ ,  $\mathbf{X}$  et  $\mathbf{X}^0$  satisfont l'équation (4.11). Dans le cas où  $S = C_m$ , l'équation (4.11) donne avec l'équation (4.19)



$$(1 - \sum_{j \in C_m} \alpha_j V_j^{C_m}) \sum_{j \in C_m} V_j^{C_m} x_j = \sum_{j \in C_m} \rho_j V_j^{C_m} + \sum_{m' \in \mathcal{M} - \{m\}} \sum_{i \in C_{m'}} \sum_{j \in C_m} (\lambda_i V_i^{C_m} - \rho_i) V_j^{C_m} x_{ij} \quad (4.20)$$

En outre, il faut noter que

$$\sum_{m' \in \mathcal{M} - \{m\}} \sum_{i \in C_{m'}} \sum_{j \in C_m} (\lambda_i V_i^{C_m} - \rho_i) V_j^{C_m} x_{ij} = \sum_{m' \in \mathcal{M} - \{m\}} \sum_{i \in C_{m'}} V_j^{C_m} \sum_{i \in C_{m'}} \rho_i R(i, m) x_{ij} \quad (4.21)$$

$$\begin{aligned} &\leq \sum_{m' \in \mathcal{M} - \{m\}} \bar{R}(m', m) \sum_{j \in C_{m'}} V_j^{C_m} \sum_{i \in C_{m'}} \rho_i x_{ij} \\ &\leq \sum_{m' \in \mathcal{M} - \{m\}} \bar{R}(m', m) \sum_{j \in C_m} V_j^{C_m} x_j, \end{aligned} \quad (4.22)$$

où l'identité (4.21) découle de (4.1) et l'inégalité (4.22) vient de (4.16).

Maintenant en combinant l'identité (4.20) avec l'inégalité (4.22) et la condition (4.12), nous avons

$$\sum_{j \in C_m} V_j^{C_m} x_j \leq \frac{1}{1 - \bar{\rho}(m)} \sum_{j \in C_m} \rho_j V_j^{C_m}, \text{ pour } m \in \mathcal{M}. \quad (4.23)$$

De (4.23), il résulte que

$$\begin{aligned} \sum_{j \in \mathcal{A}} x_j &= \sum_{m \in \mathcal{M}} \sum_{j \in C_m} x_j \\ &\leq \sum_{m \in \mathcal{M}} \sum_{j \in C_m} \frac{\rho_j V_j^{C_m}}{(1 - \bar{\rho}(m)) V_j^{C_m}} \end{aligned}$$

où  $V_j^{C_m} = \min_{j \in C_m} V_j^{C_m} \geq \min_{j \in C_m} \frac{1}{\mu_j} > 0$ , pour  $m \in \mathcal{M}$ .

Désormais, la performance LP non oisive est bornée et le Théorème 2 découle du Théorème 3.

□

Dans les paragraphes 4.3 et 4.4, nous supposons que les *MQNETs* qui font l'objet de notre analyse satisfont à l'équation (4.12).

### 4.3 Optimisation approximative des règles d'index par les lois de conservation approximatives

Nous supposons que la condition de stabilité globale (4.12) soit satisfaite et qu'elle permette d'envisager comment planifier l'ordre de passage des avions de façon optimale. Considérons une structure de coût dans laquelle les *j*-avions encourrent des coûts linéaires à un taux  $c_j$  par unité de temps dans le système. Le problème d'ordonnancement optimal consiste d'une part, à trouver une règle d'ordonnancement admissible qui minimise la somme du nombre moyen des différentes classes d'avions affectés des coefficients  $c_j$ ,

$$c_1 E[L_1] + \dots + c_N E[L_N], \quad (4.24)$$

et d'autre part, l'évaluation du coût minimal correspondant  $Z^{OPT}$ .

Ce problème a été résolu en ce qui concerne le cas particulier où le réseau est composé d'un seul serveur, c'est-à-dire  $M = 1$ . C'est en analysant ce cas particulier que Klimov (1974) a, le premier, montré que la méthode d'ordonnancement optimale est donnée par une règle *index-priorité* s'énonçant comme suit: pour chaque classe  $i$ , on peut calculer l'*index-priorité*  $\gamma_i$ , de telle manière qu'il soit possible de servir de façon optimale un client à tout moment et grâce à un index plus large. Le vecteur d'indices de priorité optimale  $\gamma = (\gamma_j)_{j \in \mathcal{A}}$  est calculé par l'algorithme de Klimov adapté présenté ci-dessous avec les arguments  $(\mathbf{c}, \mathbf{V})$ , où  $\mathbf{c} = (c_j)_{j \in \mathcal{A}}$  et  $\mathbf{V} = (V_j^S)_{j \in S, S \subseteq \mathcal{A}}$ . De ce fait, l'index  $\gamma_i$  calculé représente la décroissance maximale du taux de coût par unité de temps pour le traitement d'un *i*-avion qui se trouve en service dans le réseau.

#### Algorithme de Klimov adapté

Arguments:  $(\mathbf{c}, \mathbf{A})$ , où  $\mathbf{c} = (c_j)_{j \in \mathcal{A}}$  et  $\mathbf{A} = (A_j^S)_{j \in S, S \subseteq \mathcal{A}}$

Résultats:  $(\pi, \bar{\gamma}, \gamma)$ , où  $\pi = (\pi_1, \dots, \pi_N)$  est une permutation de  $\mathcal{A}$ ,  $\bar{\gamma} = (\bar{\gamma}(S))_{S \subseteq \mathcal{A}}$  et  $\gamma = (\gamma_1, \dots, \gamma_N)$ .

Initialisation.  $S_1 := \mathcal{A}$ ;  $\bar{\gamma}(S_1) := \min \left\{ \frac{c_i}{A_i^{S_1}}; i \in S_1 \right\}$ ;



$$\text{pick } \pi_1 \in \arg \min \left\{ \frac{c_i}{A_i^{S_1}}; i \in S_1 \right\};$$

$$\gamma_{\pi_1} := \bar{y}(S_1).$$

Itération k. Pour  $k = 2, \dots, N$ :

$$S_k := S_{k-1} - \{\pi_{k-1}\}; \bar{y}(S_k) := \min \left\{ \frac{c_i - \sum_{j=1}^{k-1} A_i^{S_j} \bar{y}(S_j)}{A_i^{S_k}} : i \in S_k \right\};$$

$$\text{pick } \pi_k \in \arg \min \left\{ \frac{c_i - \sum_{j=1}^{k-1} A_i^{S_j} \bar{y}(S_j)}{A_i^{S_k}} : i \in S_k \right\};$$

$$\gamma_{\pi_k} := \gamma_{\pi_{k-1}} + \bar{y}(S_k).$$

Itération N+1. Pour  $S \subseteq \mathcal{N}$ :  $\bar{y}(S) := 0$ , si  $S \notin \{S_1, \dots, S_N\}$ .

La modification de la règle d'index pour le cas où  $M = 1$ , cas qui correspond à notre modèle *MQNET* plus complexe, est celle qui produit des indices en exécutant l'algorithme adapté de Klimov à partir des arguments  $(\mathbf{c}, \mathbf{A})$ .  $\mathbf{A} = (A_i^S)_{i \in S, S \subseteq \mathcal{N}}$  est donné par la relation

$$A_i^S = \begin{cases} V_i^{S \cap C_m} & \text{si } i \in S \cap C_m \\ 0 & \text{sinon} \end{cases} \quad (4.25)$$

Des résultats de Klimov (1974), on déduit qu'une telle règle d'index est optimale, si les stations dans le *MQNET* sont autonomes, c'est-à-dire lorsqu'il n'y a aucun flux entre elles. En outre, le choix de la matrice  $\mathbf{A}$  dans (4.25) découle naturellement de la formulation du problème d'ordonnancement optimal donné sous les lois approximatives de conservation qui sont décrites à présent.

**Définition 1** (Lois approximatives de conservation) *Supposons que  $\mathbf{A} = (A_i^S)_{i \in S, S \subseteq \mathcal{N}}$  soit une matrice avec tous les  $A_i^S > 0$ , que  $b$  et  $\Phi: 2^{\mathcal{N}} \rightarrow R^+$  soient un ensemble de fonctions non négatives. Le vecteur de performance  $\mathbf{x}$  satisfait les lois approximatives de conservation avec les paramètres  $\mathbf{A}$ ,  $b$  et  $\Phi$ , si les conditions suivantes sont satisfaites:*

(i) *Sous une règle d'ordonnancement quelconque admissible*

$$\sum_{j \in S} A_j^S x_j \geq b(S), \text{ pour } S \subseteq \mathcal{N}. \quad (4.26)$$

(ii) *Sous une règle d'ordonnancement quelconque qui donne la priorité aux  $S$ -clients au détriment des  $S^C$ -clients.*

$$\sum_{j \in S} A_j^S x_j \geq b(S) + \Phi(S), \text{ pour } S \subseteq \mathcal{N}. \quad (4.27)$$

**Remarques:**

- 1- Dans le cas particulier où  $\Phi \equiv 0$ , la Définition 1 est réduite aux lois de conservation généralisées introduites par Bertsimas et Niño Mora (1996).
- 2- La Définition 1 diffère de la définition originelle de Glazebrook et Garbe (1997) dans la mesure où une hypothèse restrictive est nécessaire pour que la partie (ii) de la définition soit satisfaite. Cette modification est essentielle afin d'appliquer la loi de conservation approximative de Glazebrook et Garbe au modèle présenté dans ce travail.

Quand un vecteur de performance satisfait aux lois de conservation approximatives, nous pouvons résoudre de manière approximative le problème d'ordonnancement optimal sur les objectifs de performance linéaire comme dans (4.24). Exprimons (4.24) comme

$$c_1 x_1 + \dots + c_N x_N \quad (4.28)$$

en utilisant la relation (4.6). Les problèmes de ce type sont abordés via la *méthode de la région réalisable* (Bertsimas 1995). En procédant de cette façon, nous considérons que la *région de performance* réalisable par le vecteur de performance  $\mathbf{x}$  est  $\chi$  sous toutes les règles d'ordonnancement admissibles. Le problème d'ordonnancement optimal peut dès lors être formulé grâce à l'équation

$$Z^{OPT} = \min \{c_1 x_1 + \dots + c_N x_N : \mathbf{x} \in \chi\}. \quad (4.29)$$

Supposons que le vecteur de performance satisfait les lois de conservation approximatives (4.26) et (4.27). De ce fait, la région de performance  $\chi$  est contenue dans le polyèdre



$$= \left\{ x \in R_N^+ : \sum_{j \in S} A_j^S x_j \geq b(S), S \subseteq \mathcal{N} \right\},$$

qui fournit la solution programmation linéaire (LP) du problème (4.29):

$$Z^{LP} = \min \{ c_1 x_1 + \dots + c_N x_N : x \in \mathcal{P} \} \quad (4.30)$$

Nous obtenons les garanties de performance pour les règles d'index appropriées dans le Théorème 4 en construisant des solutions possibles pour (4.29) et pour le dual (4.30) comme suit:

- 1- Exécuter l'algorithme de Klimov adapté avec  $(c, A)$  comme input pour obtenir  $(\pi, \bar{y}, \gamma)$ .
- 2- La solution heuristique proposée pour le problème (4.29) est le vecteur de performance de la règle d'ordonnancement de l'index de priorité qui donne une grande priorité aux classes comportant un grand  $\gamma_i$ . La valeur atteinte par cette règle est notée  $Z^{INDEX}$ .
- 3- La solution proposée pour le dual de l'accomplissement LP (4.30) est donné par  $\bar{y}$  (Bertsimas & Niño Mora 1996a)

Afin de simplifier la présentation des résultats, nous devons supposer par la suite que la permutation  $\pi$  renvoyée par l'algorithme de Klimov est  $\pi = (1, \dots, N)$  tel que

$$\gamma_1 \leq \dots \leq \gamma_N$$

En outre, Bertsimas et Niño Mora (1996a) ont montré que la valeur possible de la solution dual  $\bar{y}$  est donnée par

$$Z^D = \gamma_1 b(\{1, \dots, N\}) + (\gamma_2 - \gamma_1) b(\{2, \dots, N\}) + \dots + (\gamma_N - \gamma_{N-1}) b(\{N\}) \quad (4.31)$$

Notons que

$$Z^D \leq Z^{LP} \leq Z^{OPT} \leq Z^{INDEX} \quad (4.32)$$

Le résultat qui reformule le résultat d'optimisation approximative originel de Glazebrook et Garbe (1997) fournit des garanties de performance additives et multiplicatives

pour la valeur  $Z^{INDEX}$  de l'heuristique index-priorité et pour la valeur  $Z^D$  de la borne inférieure.

**Théorème 4** (Garantie de performance) *Supposons que le vecteur de performance  $\mathbf{x}$  satisfait les lois de conservation approximatives (4.26) et (4.27) avec les paramètres  $\mathbf{A}$ ,  $\mathbf{b}$  et  $\Phi$ . De ce fait,*

$$(a) \quad Z^{INDEX} \leq Z^{OPT} + \varepsilon \quad (4.33)$$

et

$$Z^D \geq Z^{OPT} - \varepsilon \quad (4.34)$$

$$\text{où } \varepsilon = \gamma_1 \Phi(\{1, \dots, N\}) + (\gamma_2 - \gamma_1) \Phi(\{2, \dots, N\}) + \dots + (\gamma_N - \gamma_{N-1}) \Phi(\{N\}) \quad (4.35)$$

$$(b) \quad \text{En outre } Z^{INDEX} \leq r Z^{OPT} \quad (4.36)$$

$$\text{et } Z^D \geq \frac{1}{r} Z^{OPT} \quad (4.37)$$

$$\text{où } r = 1 + \max_{S \subseteq \mathcal{N}} \frac{\Phi(S)}{b(S)} \quad (4.38)$$

### Démonstration

(a) Bertsimas et Niño Mora (1996a) démontrent que  $\bar{y}$  est une solution du dual LP de (4.30) donné par

$$Z^{LP} = \max \sum_{S \subseteq \mathcal{N}} b(S) y(S)$$

$$\text{sujet à } \sum A_i^S y(S) \leq c_i, \text{ pour } i \in \mathcal{N}$$

$$y(S) \geq 0, \text{ pour } S \subseteq \mathcal{N},$$

ce qui satisfait la contrainte avec égalité, c'est-à-dire



$$\sum_{i \in S} A_i^S \bar{y}(S) = c_i, \text{ pour } i \in \mathcal{A}, \quad (4.39)$$

De plus, ils ont prouvé la relation

$$\gamma_i - \gamma_{i-1} = \bar{y}(S_i), \text{ pour } i \in \mathcal{A},$$

qui est satisfaite où  $S_i = \{i, i+1, \dots, N\}$ .

À présent, supposons que  $\mathbf{x}^{\text{INDEX}}$  soit le vecteur de performance correspondant à la règle d'index proposée. Nous obtenons

$$\begin{aligned} Z^{\text{INDEX}} &= \sum_{i \in \mathcal{A}} c_i x_i^{\text{INDEX}} \\ &= \sum_{i \in \mathcal{A}} \sum_{i \in S} A_i^S \bar{y}(S) x_i^{\text{INDEX}} \\ &= \sum_{S \subseteq \mathcal{A}} \bar{y}(S) \sum_{i \in S} A_i^S x_i^{\text{INDEX}} \\ &\leq \sum_{i=1}^N \bar{y}(S_i) \{b(S_i) + \Phi(S_i)\} \\ &= Z^D + \varepsilon \end{aligned} \quad (4.40)$$

et à partir de la relation (4.32), sont déduites les relations (4.33) et (4.34). La partie (b) découle des relations (4.31), (4.38) et (4.40).

□

**Remarque:** Dans le cas particulier où  $\Phi \equiv 0$ , on déduit du Théorème 4 que la règle d'index proposée est optimale (Bertsimas & Niño Mora 1996a).

Notre objectif est à présent d'établir les lois de conservation approximatives pour le modèle *MQNET* exposé. L'application du Théorème 4 fournira les garanties de performance pour une règle d'index définie et appropriée.

Pour atteindre cet objectif, nous devons utiliser les notions de pic d'intensité de trafic énoncées dans les relations (4.3) (4.4) et (4.5) qui ont été présentées dans la première partie de ce chapitre. Soient  $S \subseteq \mathcal{A}$ ,  $m \in \mu(S)$  et  $i \in C_m$  avec  $m \neq m'$ , nous définissons le  $(m, m')$ -pic

d'intensité de trafic de la classe  $i \in C_{m'}$  vers  $S \cap C_m$ , noté  $R_{m',m}(i, S \cap C_m)$ , comme le taux maximal par lequel le travail peut être transféré de la classe  $i$  dans les classes  $S \cap C_m$ , c'est-à-dire

$$R_{m',m}(i, S \cap C_m) = \mu_i \sum_{j \in S \cap C_m} P_{ij} V_j^{S \cap C_m} \quad (4.41)$$

Soit  $S'$  tel que  $S' \subseteq \mathcal{N} - C_m$ , nous pouvons désormais définir le  $m$ -pic d'intensité de trafic de  $S'$  dans  $S \cap C_m$  noté  $\bar{R}_m(S', S \cap C_m)$ , comme le taux maximal par lequel le travail peut être transféré des classes dans  $S'$  vers des classes dans  $S \cap C_m$ , c'est-à-dire,

$$\bar{R}_m(S', S \cap C_m) = \sum_{m' \in \mu(S')} \max_{i \in S' \cap C_{m'}} R_{m',m}(i, S \cap C_m).$$

**Théorème 5** (Lois de conservation approximatives pour le modèle MQNET). *Sous la condition globale de stabilité (4.12), le modèle MQNET satisfait les lois de conservation approximatives avec les paramètres  $\mathbf{A}$ ,  $\mathbf{b}$  et  $\Phi$  donnés par les relations suivantes:*

$$A_i^S = \begin{cases} V_i^{S \cap C_m}, & \text{si } i \in S \cap C_m \\ 0, & \text{sinon} \end{cases}$$

$$b(S) = \sum_{m \in \mu(S)} \hat{b}(S \cap C_m);$$

et 
$$\Phi(S) = \sum_{m \in \mu(S)} \frac{\bar{R}_m(\mathcal{N} - C_m, S \cap C_m)}{1 - \rho^o(S \cap C_m) - \bar{R}_m(\mathcal{N} - C_m, S \cap C_m)} \hat{b}(S \cap C_m)$$

où 
$$\hat{b}(S) = \frac{1}{1 - \rho^o(S)} \sum_{j \in S} \rho_j V_j^S$$

**Démonstration:**

Soient  $S \subseteq \mathcal{N}$ ,  $m \in \mu(S)$ . Des contraintes de décomposition du travail (4.11) et de la non négativité des mesures de performance, nous déduisons que sous une règle admissible (garantie stable par le Théorème 2)



$$\sum_{j \in S \cap C_m} V_j^{S \cap C_m} x_j \geq \frac{\sum \rho_j V_j^{S \cap C_m}}{1 - \rho^0(S \cap C_m)} = \hat{b}(S \cap C_m) \quad (4.43)$$

En additionnant les deux membres de l'inégalité (4.43) sur  $m \in \mu(S)$ , nous inférons la relation (4.26) en reprenant les paramètres  $A$ ,  $b$  et  $\Phi$  donnés dans le Théorème 5.

Pour obtenir la relation (4.27), considérons des règles  $\pi$  qui donnent priorité aux  $S$ -avions sur  $S^C$ -avions.

Sous  $\pi$ , il résulte de la définition des variables de performance dans les relations (4.7) et (4.8) que

$$x_{ij} = 0, \quad \text{si } i \in S^C \cap C_m, \quad j \in S \cap C_m, \quad m \in \mu(S) \quad (4.44)$$

et que

$$x_j^{0m} = 0, \quad \text{si } j \in S \cap C_m \quad (4.45)$$

En utilisant (4.43) et (4.44) dans (4.11), nous obtenons

$$\{1 - \rho^0(S \cap C_m)\} \sum_{j \in S \cap C_m} V_j^{S \cap C_m} x_j = \sum_{j \in S \cap C_m} \rho_j V_j^{S \cap C_m} + \sum_{i \notin C_m} \sum_{j \in S \cap C_m} (\lambda_i V_i^{S \cap C_m} - \rho_i) V_j^{S \cap C_m} x_{ij} \quad (4.46)$$

À présent, en utilisant (4.1), le second terme de la partie droite de (4.46) peut être réécrit comme

$$\begin{aligned} & \sum_{m' \neq m} \sum_{i \in C_{m'}} \sum_{k \in S \cap C_m} (\mu_i V_k^{S \cap C_m} p_{ik}) V_j^{S \cap C_m} \rho_i x_{ij} \\ & \leq \left\{ \sum_{m' \neq m} \max_{i \in C_{m'}} R_{m',m}(i, S \cap C_m) \right\} \left\{ \sum_{j \in S \cap C_m} V_j^{S \cap C_m} \left( \sum_{k \in C_{m'}} \rho_k x_{kj} \right) \right\} \end{aligned} \quad (4.47)$$

$$\leq \bar{R}_m(\mathcal{N} - C_m, S \cap C_m) \sum_{j \in S \cap C_m} V_j^{S \cap C_m} x_j \quad (4.48)$$

Nous utilisons la relation (4.41) pour déduire les relations (4.47) et (4.9); et la relation (4.42) pour déduire la relation (4.48). À partir de la relation (4.46), de la relation (4.48) et de la condition de stabilité (4.12), nous déduisons sous  $\pi$  que

$$\sum_{j \in S \cap C_m} V_j^{S \cap C_m} x_j \leq \frac{\sum_{j \in S \cap C_m} \rho_j V_j^{S \cap C_m}}{1 - \rho^0(S \cap C_m) - \bar{R}(\mathcal{N} - C_m, S \cap C_m)} \quad (4.49)$$

Notons que (4.12) garantit que le dénominateur dans (4.49) soit positif. À présent, en additionnant chaque membre de l'inégalité (4.49) sur  $m \in \mu(S)$  et en utilisant la relation (4.43), nous inférons la relation (4.27) en reprenant les paramètres  $A$ ,  $b$  et  $\Phi$  donnés ci-dessus.  $\square$

C'est de multiples façons que les Théorèmes 4 et 5 peuvent être exploités dans le but de donner des garanties de performance pour la règle d'index développée ci-dessus. Nous devons nous contenter de deux résultats. Pour le réseau, on définit  $r$  comme

$$r = 1 + \max_{S \subseteq \mathcal{N}} \frac{\Phi(S)}{b(S)}$$

avec  $b$  et  $\Phi$  donnés dans l'énoncé du Théorème 5. À partir de la définition des quantités en action, on peut établir que

$$r \leq 1 + \max_{m \in \mathcal{M}} \frac{\bar{R}_m(\mathcal{N} - C_m, C_m)}{1 - \bar{\rho}(m)}$$

et que  $r$  est une mesure naturelle du degré de connexion des stations dans le réseau à lourd trafic.

**Théorème 6** (Garantie de performance pour la règle d'index)

$$Z^{INDEX} \leq r Z^{OPT}$$

Dans le Théorème 6, nous exploitons la garantie de performance afin d'établir l'optimalisation asymptotique pour la règle d'index dans une limite quand le flux entre stations devient élevé.



### Corollaire 1 (Optimalisation asymptotique)

$$\frac{Z^{INDEX}}{Z^{OPT}} \rightarrow 1 \quad \text{comme} \quad \max_{S(i) \neq S(j)} p_{ij} \rightarrow 0$$

#### Démonstration

Le résultat résulte immédiatement du Théorème 6 dans lequel, pour tout  $S$  et  $m$ ,

$$\bar{R}_m(\mathcal{N} - C_m, S \cap C_m) \rightarrow 0 \quad \text{quand} \quad \max_{S(i) \neq S(j)} p_{ij} \rightarrow 0$$

et c'est pourquoi

$$r \rightarrow 1 \quad \text{quand} \quad \max_{S(i) \neq S(j)} p_{ij} \rightarrow 0$$

□

## 4.4 Analyse de performance des règles de priorité par les lois de conservation approximatives

Notre objectif dans cette dernière partie est d'analyser la performance de certaines règles de priorité  $\pi$  qui représente un intérêt pour le *MQNET*.  $\mathbf{x}^\pi$  représente la performance et  $Z^\pi$  est l'objectif. En analysant la performance nous poursuivrons deux objectifs: d'une part, l'évaluation de  $Z^\pi$  et d'autre part, le développement des garanties de performance pour  $\pi$  sous la forme de bornes pour  $Z^\pi - Z^{OPT}$ .

Soient  $S_k^\pi = \{\pi_k, \dots, \pi_N\}$  les  $(N - k + 1)$  plus grandes classes de priorité sous  $\pi$ . Les paramètres  $\mathbf{A}$ ,  $\mathbf{b}$  et  $\Phi$  sont définis dans le Théorème 5. Nous introduisons  $\mathbf{y}^\pi$  comme la solution du système d'équations linéaires

$$\sum_{i=1}^k A_k^{S_i^\pi} y_i^\pi = c_k, \quad \text{si } 1 \leq k \leq N. \quad (4.50)$$

Dans le Théorème 7, nous écrivons  $u^+ = \max(u, 0)$  et  $u^- = \max(-u, 0)$ .

**Théorème 7** (Évaluation de  $Z^\pi$ ) Sous la condition de stabilité globale (4.12)

$$-\sum_{k=1}^N (y_k^\pi)^- \Phi(S_k^\pi) \leq Z^\pi - \sum_{k=1}^N y_k^\pi b(S_k^\pi) \leq \sum_{k=1}^N (y_k^\pi)^+ \Phi(S_k^\pi) \quad (4.51)$$

### Démonstration

La règle  $\pi$  donne la priorité aux  $S_k^\pi$  - avions sur les autres pour tout  $k$ ,  $1 \leq k \leq N$ .

Il résulte du Théorème 5 que

$$b(S_k^\pi) \leq \sum_{i=k}^N A_i^{S_k^\pi} x_i^\pi \leq b(S_k^\pi) + \Phi(S_k^\pi), \text{ si } 1 \leq k \leq N. \quad (4.52)$$

Mais il résulte de la relation (4.50) que

$$Z^\pi = \sum_{k=1}^N c_k x_k^\pi = \sum_{k=1}^N y_k^\pi \left( \sum_{i=k}^N A_i^{S_k^\pi} x_i^\pi \right) \quad (4.53)$$

Le résultat est déduit des relations (4.52) et (4.53).

□

Avant de procéder au développement des bornes interprétables pour  $Z^\pi - Z^{\text{OPT}}$ , nous envisageons une modification de la matrice de feed-back  $\mathbf{P} = (p_{ij})$ . La matrice  $\tilde{\mathbf{P}}$  modifie  $\mathbf{P}$  en faisant une estimation des flux entre des stations distinctes grâce aux départs des clients du système, c'est-à-dire,

$$\tilde{\mathbf{P}} = \begin{cases} p_{ij} & \text{si } s(i) = s(j) \\ 0 & \text{sinon} \end{cases}$$

avec  $\tilde{p}_{i0} = 1 - \sum_{j \in A} \tilde{p}_{ij}$ . Notons que le mécanisme interne de feed-back pour chaque station n'est pas affecté. Nous devons utiliser le symbole  $\sim$  pour noter les quantités relatives au système modifié. De cette façon, nous remplaçons  $\mathbf{P}$  par  $\tilde{\mathbf{P}}$ , mais les paramètres  $\alpha$  et  $\mu$  restent inchangés.



Il est aisé de vérifier que la A-matrice définie dans le Théorème 5 n'est pas affectée par cette modification. Il en résulte que les indices Gittins obtenus à partir de l'algorithme de Klimov sont aussi inchangés tout comme l'est  $y^\pi$  dans la relation (4.50) présentée ci-dessus.

À partir du Théorème 5, nous obtenons

$$\tilde{b}(S) = \sum_{m \in \mu(S)} \left\{ \frac{\sum_{j \in S \cap C_m} \tilde{\rho}_j V_j^{S \cap C_m}}{1 - \sum_{j \in S \cap C_m} \alpha_j V_j^{S \cap C_m}} \right\} \quad (4.55)$$

où  $\tilde{\rho}_j = \frac{\tilde{\lambda}_j}{\mu_j}$ ,  $j \in \mathcal{N}$ , avec  $\tilde{\lambda}$  qui représente le vecteur des taux totaux d'arrivées du système

modifié. Notons que, par construction  $\tilde{\rho}_j \leq \rho_j$ ,  $j \in \mathcal{N}$ . Nous écrivons

$$\Delta(S) = b(S) - \tilde{b}(S) = \sum_{m \in \mu(S)} \left\{ \frac{\sum_{j \in S \cap C_m} (\rho_j - \tilde{\rho}_j) V_j^{S \cap C_m}}{1 - \rho^0(S \cap C_m)} \right\} \quad (4.56)$$

Les quantités  $\Delta(S)$  sont les mesures naturelles de la non-connexion du réseau de départ. Bien qu'il n'existe aucun flux entre les stations dans le réseau modifié, il résulte du Théorème 5 que  $\Phi \equiv 0$ . De ce fait, du Théorème 7, on déduit que

$$\sum_{k=1}^N y_k^\pi \tilde{b}(S_k^\pi) = \tilde{Z}^\pi,$$

qui représente le coût encouru quand on contrôle le réseau modifié par  $\pi$ .

La suite est une conséquence immédiate du Théorème 7 et des observations ci-dessus.

## Corollaire 2

$$\Delta_1(\pi; \mathbf{P}, \tilde{\mathbf{P}}) \leq Z^\pi - \tilde{Z}^\pi \leq \Delta_2(\pi; \mathbf{P}, \tilde{\mathbf{P}}),$$

où

$$\Delta_1(\pi; \mathbf{P}, \tilde{\mathbf{P}}) = -\sum_{k=1}^N (y_k^\pi)^- \Phi(S_k^\pi) + \sum_{k=1}^N y_k^\pi \Delta(S_k^\pi),$$

et

$$\Delta_2(\pi; \mathbf{P}, \tilde{\mathbf{P}}) = -\sum_{k=1}^N (y_k^\pi)^+ \Phi(S_k^\pi) + \sum_{k=1}^N y_k^\pi \Delta(S_k^\pi).$$

**Corollaire 3** (Borne de suboptimisation pour  $\pi$ )

$$\begin{aligned} Z^\pi - Z^{OPT} &\leq (Z^{INDEX} - Z^{OPT}) + (\tilde{Z}^\pi - \tilde{Z}^{INDEX}) \\ &\quad + \Delta_2(\pi; \mathbf{P}, \tilde{\mathbf{P}}) - \Delta_2(INDEX; \mathbf{P}, \tilde{\mathbf{P}}). \end{aligned}$$

**Démonstration**

$$\begin{aligned} Z^\pi - Z^{OPT} &= (Z^\pi - \tilde{Z}^\pi) + (\tilde{Z}^\pi - \tilde{Z}^{INDEX}) \\ &\quad + (\tilde{Z}^{INDEX} - Z^{INDEX}) + (Z^{INDEX} - Z^{OPT}). \end{aligned} \tag{4.57}$$

Le corollaire 2 est à présent utilisé pour borner le premier et le troisième terme de la partie droite de (3.57).

□

**Remarque:**

Le corollaire 3 résout essentiellement le problème des garanties de performance pour la règle de priorité  $\pi$ . En effet,

- (i) la borne pour  $(Z^{INDEX} - Z^{OPT})$  est immédiatement disponible dans les Théorèmes 4 et 5. Cette borne agrégée avec  $\Delta_2(\pi; \mathbf{P}, \tilde{\mathbf{P}}) - \Delta_1(INDEX; \mathbf{P}, \tilde{\mathbf{P}})$  dans le corollaire 3, donne une mesure naturelle du degré de non connexion du réseau;
- (ii) Par la construction exposée ci-dessus,  $\tilde{Z}^\pi - \tilde{Z}^{INDEX}$  doit être interprétée de la même façon que la suboptimisation de la règle  $\pi$ , quand elle est appliquée au système modifié (comportant des stations autonomes non connectées) pour lequel notre règle d'index est reconnue comme optimale par Klimov (1974). De cette façon, une borne



basée sur l'index pour  $\tilde{Z}^\pi - \tilde{Z}^{INDEX}$  est établie dans le travail de Glazebrook et Garbe (1996) sous la forme d'une mesure naturelle de l'extension pour laquelle  $\pi$  s'écarte de la règle d'index.

L'application de (i) et de (ii) au Corollaire 3 fournit une garantie de performance pour  $\pi$  qui combine une mesure du degré de non connexion du réseau avec une mesure de l'extension pour laquelle  $\pi$  s'écarte de la règle naturelle d'index.

## Chapitre 5

### MQNET: étude du cas de la navigation aérienne

Dans ce chapitre, nous appliquons au cas de la navigation aérienne les résultats théoriques obtenus au chapitre précédent. Un mémoire n'est pas suffisant pour considérer l'ensemble du réseau européen. C'est pourquoi nous avons choisi d'étudier le cas de trois aéroports fictifs (1, 2 et 3) et de leur espace aérien.

#### 5.1 Le modèle MQNET de la navigation aérienne

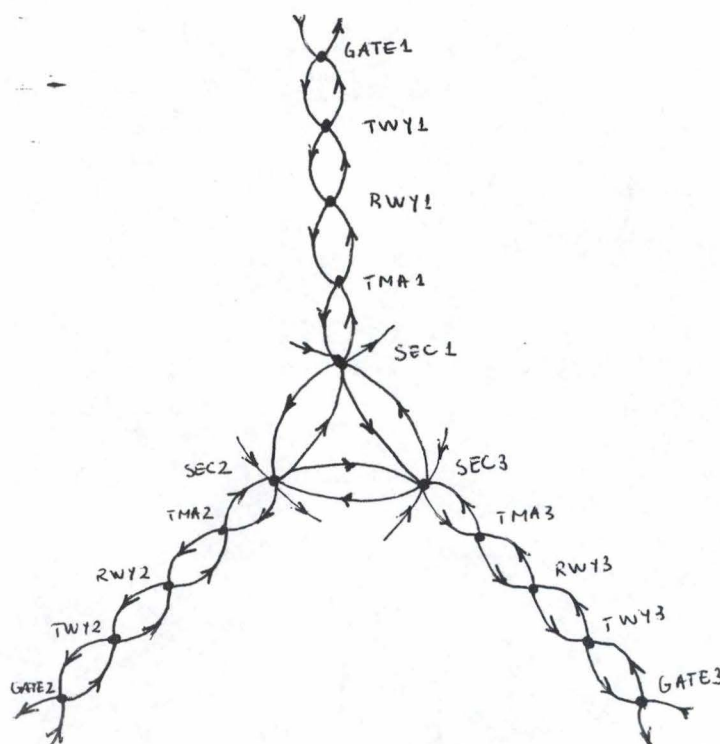


Fig. 6-1 Schéma du réseau

Le réseau de files d'attente de la figure 6-1 est ouvert. Il comporte 15 stations de service ( $M=15$ ). L'ensemble des stations de service est  $\mathcal{M}$ .



$$\mathcal{M} = \{GATE1, GATE2, GATE3, TWY1, TWY2, TWY3, RWY1, RWY2, RWY3, TMA1, TMA2, TMA3, SEC1, SEC2, SEC3\}$$

$\mathcal{N}$  étant l'ensemble de toutes les classes d'avions du réseau, une station  $m \in \mathcal{M}$  dessert un ensemble de classes d'avions  $C_m \subseteq \mathcal{N}$  où  $C_1, \dots, C_M$  constituent une partition de  $\mathcal{N}$ . Nous répartissons les classes d'avions de la façon suivante:

- une station de service de type *GATE* dessert deux types d'avions: d'une part les avions qui embarquent des passagers (notés *emb*) et d'autre part ceux qui débarquent des passagers (notés *deb*)

$$C_{GATE} = \{emb, deb\}$$

- une station de service de type *TWY* dessert deux types d'avions: d'une part les avions qui se dirigent vers le *GATE* (notés *twy→gate*) et d'autre part ceux qui vont vers la piste *RWY* (notés *twy→rwy*)

$$C_{TWY} = \{twy \rightarrow gate, twy \rightarrow rwy\}$$

- une station de service de type *RWY* dessert deux types d'avions: d'une part les avions qui se dirigent vers le *TWY* (notés *rwy→twy*) et d'autre part ceux qui vont vers le *TMA* (notés *rwy→tma*)

$$C_{RWY} = \{rwy \rightarrow twy, rwy \rightarrow tma\}$$

- une station de service de type *TMA* dessert deux types d'avions: d'une part les avions qui empruntent la piste (*RWY*) pour atterrir (notés *tma→rwy*) et d'autre part ceux qui se dirigent vers le secteur (*SEC*) (notés *tma→sec*)

$$C_{TMA} = \{tma \rightarrow rwy, tma \rightarrow sec\}$$

- une station de service de type *SEC* dessert trois types d'avions: les avions qui quittent le réseau (notés *sec→o*), ceux qui se dirigent vers le *TMA* (notés *sec→tma*), ceux qui vont vers un autre secteur (notés *sec<sub>a</sub>→sec<sub>b</sub>*).

$$C_{SEC} = \{sec \rightarrow o, sec \rightarrow tma, sec_a \rightarrow sec_b\}.$$

En tenant compte de toutes les stations de service du réseau, nous obtenons

$$\begin{aligned} \mathcal{N} = \{ & deb1, deb2, deb3, emb1, emb2, emb3, \\ & twy1 \rightarrow rwy1, twy2 \rightarrow rwy2, twy3 \rightarrow rwy3, twy1 \rightarrow gate1, twy2 \rightarrow gate2, twy3 \rightarrow gate3, \\ & dec1, dec2, dec3, att1, att2, att3, \\ & tma1 \rightarrow rwy1, tma2 \rightarrow rwy2, tma3 \rightarrow rwy3, tma1 \rightarrow sec1, tma2 \rightarrow sec2, tma3 \rightarrow sec3, \\ & sec1 \rightarrow o, sec1 \rightarrow sec2, sec1 \rightarrow sec3, sec1 \rightarrow tma1, \\ & sec2 \rightarrow o, sec2 \rightarrow sec1, sec2 \rightarrow sec3, sec2 \rightarrow tma2, \\ & sec3 \rightarrow o, sec3 \rightarrow sec1, sec3 \rightarrow sec2, sec3 \rightarrow tma3 \} \end{aligned}$$

Le nombre total des classes d'avions du réseau est  $N = 36$ . Chaque station de service dessert un nombre restreint de classes d'avions.

$$\begin{aligned} C_{GATE1} &= \{deb1, emb1\} \\ C_{GATE2} &= \{deb2, emb2\} \\ C_{GATE3} &= \{deb3, emb3\} \\ C_{TWY1} &= \{twy1 \rightarrow rwy1, twy1 \rightarrow gate1\} \\ C_{TWY2} &= \{twy2 \rightarrow rwy2, twy2 \rightarrow gate2\} \\ C_{TWY3} &= \{twy3 \rightarrow rwy3, twy3 \rightarrow gate3\} \\ C_{RWY1} &= \{dec1, att1\} \\ C_{RWY2} &= \{dec2, att2\} \\ C_{RWY3} &= \{dec3, att3\} \\ C_{TMA1} &= \{tma1 \rightarrow rwy1, tma1 \rightarrow sec1\} \\ C_{TMA2} &= \{tma2 \rightarrow rwy2, tma2 \rightarrow sec2\} \\ C_{TMA3} &= \{tma3 \rightarrow rwy3, tma3 \rightarrow sec3\} \\ C_{SEC1} &= \{sec1 \rightarrow O, sec1 \rightarrow sec2, sec1 \rightarrow sec3, sec1 \rightarrow tma1\} \\ C_{SEC2} &= \{sec2 \rightarrow O, sec2 \rightarrow sec1, sec2 \rightarrow sec3, sec2 \rightarrow tma2\} \\ C_{SEC3} &= \{sec3 \rightarrow O, sec3 \rightarrow sec1, sec3 \rightarrow sec2, sec3 \rightarrow tma3\} \end{aligned}$$

Les avions venant de l'extérieur entrent dans le réseau par les *GATEs* et les *SECs*. Le réseau comporte ainsi 6 portes d'entrée: *GATE1*, *GATE2*, *GATE3*, *SEC1*, *SEC2* et *SEC3*. Ces stations de service sont également les seuls passages par lesquels les avions peuvent quitter le réseau.

### Taux d'arrivée des *i*-avions dans le réseau

Le processus d'arrivée des *i*-avions dans le réseau est de Poisson de taux  $\alpha_i$ . Les valeurs des  $\alpha_i$  susceptibles d'être non nulles sont:  $\alpha_{emb1}$ ,  $\alpha_{emb2}$ ,  $\alpha_{emb3}$ ,  $\alpha_{sec1 \rightarrow sec2}$ ,  $\alpha_{sec1 \rightarrow sec3}$ ,  $\alpha_{sec1 \rightarrow tma1}$ ,  $\alpha_{sec2 \rightarrow sec1}$ ,  $\alpha_{sec2 \rightarrow sec3}$ ,  $\alpha_{sec2 \rightarrow tma2}$ ,  $\alpha_{sec3 \rightarrow sec1}$ ,  $\alpha_{sec3 \rightarrow sec2}$ ,  $\alpha_{sec3 \rightarrow tma3}$ .



Les autres valeurs des  $\alpha_i$  sont nulles, à savoir  $\alpha_{deb1}$ ,  $\alpha_{deb2}$ ,  $\alpha_{deb3}$ ,  $\alpha_{twy1 \rightarrow rwy1}$ ,  $\alpha_{twy2 \rightarrow rwy2}$ ,  $\alpha_{twy3 \rightarrow rwy3}$ ,  $\alpha_{twy1 \rightarrow gate1}$ ,  $\alpha_{twy2 \rightarrow gate2}$ ,  $\alpha_{twy3 \rightarrow gate3}$ ,  $\alpha_{dec1}$ ,  $\alpha_{dec2}$ ,  $\alpha_{dec3}$ ,  $\alpha_{att1}$ ,  $\alpha_{att2}$ ,  $\alpha_{att3}$ ,  $\alpha_{tma1 \rightarrow rwy1}$ ,  $\alpha_{tma2 \rightarrow rwy2}$ ,  $\alpha_{tma3 \rightarrow rwy3}$ ,  $\alpha_{tma1 \rightarrow sec1}$ ,  $\alpha_{tma2 \rightarrow sec2}$ ,  $\alpha_{tma3 \rightarrow sec3}$ ,  $\alpha_{sec1 \rightarrow o}$ ,  $\alpha_{sec2 \rightarrow o}$ ,  $\alpha_{sec3 \rightarrow o}$ .

### Matrice des probabilités de transition

$$P = (p_{ij})_{i,j \in A}$$

$p_{ij}$  est la probabilité qu'un avion passe de la classe  $i$  à la classe  $j$ . Notons que toutes les transitions ne sont pas possibles. Une transition qui n'est pas possible a une probabilité nulle. Les transitions indubitables ont une probabilité égale à 1.

Les transitions indubitables sont:

$$\begin{aligned} p_{emb \ twy \rightarrow rwy} &= p_{twy \rightarrow rwy \ dec} \\ &= p_{dec \ tma \rightarrow sec} \\ &= p_{att \ twy \rightarrow gate} \\ &= p_{tma \rightarrow rwy \ att} \\ &= p_{sec \rightarrow tma \ tma \rightarrow rwy} \\ &= 1 \end{aligned}$$

Les probabilités des transitions qui sont susceptibles d'être non nulles sont:  $p_{deb \ emb}$ ,  $p_{deb \ twy \rightarrow rwy}$ ,  $p_{twy \rightarrow gate \ deb}$ ,  $p_{twy \rightarrow gate \ emb}$ ,  $p_{tma \rightarrow sec \ sec \rightarrow o}$ ,  $p_{tma \rightarrow sec \ sec \rightarrow sec}$ ,  $p_{tma \rightarrow sec \ sec \rightarrow secc}$ .

Toutes les autres probabilités de transitions sont nulles. Ce sont des transitions élémentaires impossibles.

La somme des probabilités d'une ligne de la matrice  $P$  étant égale à 1, nous obtenons:

$$p_{deb1 \ emb1} + p_{deb1 \ twy1 \rightarrow rwy1} + p_{deb1 \rightarrow o} = 1$$

$$p_{deb2 \ emb2} + p_{deb2 \ twy2 \rightarrow rwy2} + p_{deb2 \rightarrow o} = 1$$

$$p_{deb3 \ emb3} + p_{deb3 \ twy3 \rightarrow rwy3} + p_{deb3 \rightarrow o} = 1$$

$$p_{twy1 \rightarrow gate1 \ emb1} + p_{twy1 \rightarrow gate1 \ deb1} = 1$$

$$p_{twy2 \rightarrow gate2 \ emb2} + p_{twy2 \rightarrow gate2 \ deb2} = 1$$

$$p_{twy3 \rightarrow gate3 \ emb3} + p_{twy3 \rightarrow gate3 \ deb3} = 1$$

$$p_{tma1 \rightarrow sec1 \ sec1 \rightarrow o} + p_{tma1 \rightarrow sec1 \ sec1 \rightarrow sec2} + p_{tma1 \rightarrow sec1 \ sec1 \rightarrow sec3} = 1$$

$$P_{tma2 \rightarrow sec2 \ sec2 \rightarrow o} + P_{tma2 \rightarrow sec2 \ sec2 \rightarrow sec1} + P_{tma2 \rightarrow sec2 \ sec2 \rightarrow sec3} = 1$$

$$P_{tma3 \rightarrow sec3 \ sec3 \rightarrow o} + P_{tma3 \rightarrow sec3 \ sec3 \rightarrow sec1} + P_{tma3 \rightarrow sec3 \ sec3 \rightarrow sec2} = 1$$

$$P_{sec1 \rightarrow sec2 \ sec2 \rightarrow o} + P_{sec1 \rightarrow sec2 \ sec2 \rightarrow sec3} + P_{sec1 \rightarrow sec2 \ sec2 \rightarrow tma2} = 1$$

$$P_{sec1 \rightarrow sec3 \ sec3 \rightarrow o} + P_{sec1 \rightarrow sec3 \ sec3 \rightarrow sec2} + P_{sec1 \rightarrow sec3 \ sec3 \rightarrow tma3} = 1$$

$$P_{sec2 \rightarrow sec1 \ sec1 \rightarrow o} + P_{sec2 \rightarrow sec1 \ sec1 \rightarrow sec3} + P_{sec2 \rightarrow sec1 \ sec1 \rightarrow tma1} = 1$$

$$P_{sec2 \rightarrow sec3 \ sec3 \rightarrow o} + P_{sec2 \rightarrow sec3 \ sec3 \rightarrow sec1} + P_{sec2 \rightarrow sec2 \ sec3 \rightarrow tma3} = 1$$

$$P_{sec3 \rightarrow sec1 \ sec1 \rightarrow o} + P_{sec3 \rightarrow sec1 \ sec1 \rightarrow sec2} + P_{sec3 \rightarrow sec1 \ sec1 \rightarrow tma1} = 1$$

### Taux d'arrivée total des $j$ -avions $\lambda_j$

Par définition, le taux d'arrivée total des  $j$ -avions est donné par  $\lambda_j = \alpha_j + \sum_{i \in \mathcal{A}} \lambda_i p_{ij}$ , pour  $j \in \mathcal{A}$ .

Calculons ce taux pour les différentes classes d'avions du réseau en fonction des probabilités de transition.

$$\begin{aligned} \lambda_{deb} &= \alpha_{deb} + \sum_{i \in \mathcal{A}} \lambda_i p_{i deb} \\ &= \sum_{i \in \mathcal{A}} \lambda_i p_{i deb} \quad \text{car } \alpha_{deb} = 0 \\ &= \lambda_{twy \rightarrow gate} p_{twy \rightarrow gate deb} \quad \text{car seule } p_{twy \rightarrow gate deb} \neq 0. \end{aligned}$$

Les autres transitions vers la classe  $deb$  sont impossibles, donc de probabilité nulle.

$$\begin{aligned} \lambda_{emb} &= \alpha_{emb} + \sum_{i \in \mathcal{A}} \lambda_i p_{i emb} \\ &= \alpha_{emb} + \lambda_{deb} p_{deb emb} + \lambda_{twy \rightarrow gate} p_{twy \rightarrow gate emb} \end{aligned}$$

$$\begin{aligned} \lambda_{twy \rightarrow gate} &= \alpha_{twy \rightarrow gate} + \sum_{i \in \mathcal{A}} \lambda_i p_{i twy \rightarrow gate} \\ &= \lambda_{att} p_{att twy \rightarrow gate} \\ &= \lambda_{att} \end{aligned}$$

$p_{att twy \rightarrow gate} = 1$  dans la mesure où on suppose que tous les avions qui atterrissent se dirigent vers les portes en empruntant le taxiway.

$$\begin{aligned} \lambda_{twy \rightarrow rwy} &= \alpha_{twy \rightarrow rwy} + \sum_{i \in \mathcal{A}} \lambda_i p_{i twy \rightarrow rwy} \\ &= \lambda_{emb} p_{emb twy \rightarrow rwy} + \lambda_{deb} p_{deb twy \rightarrow rwy} \end{aligned}$$



$$\begin{aligned}
\lambda_{dec} &= \alpha_{dec} + \sum_{i \in \mathcal{A}} \lambda_i p_{i dec} \\
&= \lambda_{rwy \rightarrow rwy} p_{rwy \rightarrow rwy dec} \\
&= \lambda_{rwy \rightarrow rwy}
\end{aligned}$$

$p_{rwy \rightarrow rwy dec} = 1$  dans la mesure où on suppose que tous les avions qui se trouvent sur le taxiway et qui se dirigent vers la piste décollent.

$$\begin{aligned}
\lambda_{att} &= \alpha_{att} + \sum_{i \in \mathcal{A}} \lambda_i p_{i att} \\
&= \lambda_{tma \rightarrow rwy} p_{tma \rightarrow rwy att} \\
&= \lambda_{tma \rightarrow rwy}
\end{aligned}$$

$$\begin{aligned}
\lambda_{tma \rightarrow rwy} &= \alpha_{tma \rightarrow rwy} + \sum_{i \in \mathcal{A}} \lambda_i p_{i tma \rightarrow rwy} \\
&= \lambda_{sec \rightarrow tma} p_{sec \rightarrow tma tma \rightarrow rwy} \\
&= \lambda_{sec \rightarrow tma}
\end{aligned}$$

$$\begin{aligned}
\lambda_{tma \rightarrow sec} &= \alpha_{tma \rightarrow sec} + \sum_{i \in \mathcal{A}} \lambda_i p_{i tma \rightarrow sec} \\
&= \lambda_{dec} p_{dec tma \rightarrow sec} \\
&= \lambda_{dec}
\end{aligned}$$

$$\begin{aligned}
\lambda_{sec1 \rightarrow tma1} &= \alpha_{sec1 \rightarrow tma1} + \sum_{i \in \mathcal{A}} \lambda_i p_{i sec1 \rightarrow tma1} \\
&= \alpha_{sec \rightarrow tma} + \lambda_{sec2 \rightarrow sec1} p_{sec2 \rightarrow sec1 sec1 \rightarrow tma1} + \lambda_{sec3 \rightarrow sec1} p_{sec3 \rightarrow sec1 sec1 \rightarrow tma1}
\end{aligned}$$

$$\begin{aligned}
\lambda_{sec1 \rightarrow o} &= \alpha_{sec1 \rightarrow o} + \sum_{i \in \mathcal{A}} \lambda_i p_{i sec1 \rightarrow o} \\
&= \lambda_{tma \rightarrow sec1} p_{tma \rightarrow sec1 sec1 \rightarrow o} + \lambda_{sec2 \rightarrow sec1} p_{sec2 \rightarrow sec1 sec1 \rightarrow o} + \lambda_{sec3 \rightarrow sec1} p_{sec3 \rightarrow sec1 sec1 \rightarrow o}
\end{aligned}$$

$$\begin{aligned}
\lambda_{sec1 \rightarrow sec2} &= \alpha_{sec1 \rightarrow sec2} + \sum_{i \in \mathcal{A}} \lambda_i p_{i sec1 \rightarrow sec2} \\
&= \alpha_{sec1 \rightarrow sec2} + \lambda_{tma1 \rightarrow sec1} p_{tma1 \rightarrow sec1 sec1 \rightarrow sec2} + \lambda_{sec3 \rightarrow sec1} p_{sec3 \rightarrow sec1 sec1 \rightarrow sec2}
\end{aligned}$$

En tenant compte de toutes les classes d'avions du réseau, il vient:

$$\lambda_{deb1} = \lambda_{rwy1 \rightarrow gate1} p_{rwy1 \rightarrow gate1 deb1}$$

$$\lambda_{emb1} = \alpha_{emb1} + \lambda_{deb1} p_{deb1 emb1} + \lambda_{rwy1 \rightarrow gate1} p_{rwy1 \rightarrow gate1 emb1}$$

$$\lambda_{rwy1 \rightarrow gate1} = \lambda_{att1}$$

$$\lambda_{rwy1 \rightarrow rwy1} = \lambda_{emb1} p_{emb1 rwy1 \rightarrow rwy1} + \lambda_{deb1} p_{deb1 rwy1 \rightarrow rwy1}$$

$$\lambda_{dec1} = \lambda_{rwy1 \rightarrow rwy1}$$

$$\lambda_{att1} = \lambda_{ima1 \rightarrow rwy1}$$

$$\lambda_{ima1 \rightarrow rwy1} = \lambda_{sec1 \rightarrow ima1}$$

$$\lambda_{ima1 \rightarrow sec1} = \lambda_{dec1}$$

$$\lambda_{sec1 \rightarrow ima1} = \alpha_{sec \rightarrow ima1} + \lambda_{sec2 \rightarrow sec1} P_{sec2 \rightarrow sec1} P_{sec1 \rightarrow ima1} + \lambda_{sec3 \rightarrow sec1} P_{sec3 \rightarrow sec1} P_{sec1 \rightarrow ima1}$$

$$\lambda_{sec1 \rightarrow o} = \lambda_{ima1 \rightarrow sec1} P_{ima1 \rightarrow sec1} P_{sec1 \rightarrow o} + \lambda_{sec2 \rightarrow sec1} P_{sec2 \rightarrow sec1} P_{sec1 \rightarrow o} + \lambda_{sec3 \rightarrow sec1} P_{sec3 \rightarrow sec1} P_{sec1 \rightarrow o}$$

$$\lambda_{sec1 \rightarrow sec2} = \alpha_{sec1 \rightarrow sec2} + \lambda_{ima1 \rightarrow sec1} P_{ima1 \rightarrow sec1} P_{sec1 \rightarrow sec2} + \lambda_{sec3 \rightarrow sec1} P_{sec3 \rightarrow sec1} P_{sec1 \rightarrow sec2}$$

$$\lambda_{sec1 \rightarrow sec3} = \alpha_{sec1 \rightarrow sec3} + \lambda_{ima1 \rightarrow sec1} P_{ima1 \rightarrow sec1} P_{sec1 \rightarrow sec3} + \lambda_{sec2 \rightarrow sec1} P_{sec2 \rightarrow sec1} P_{sec1 \rightarrow sec3}$$

$$\lambda_{deb2} = \lambda_{rwy2 \rightarrow gate2} P_{rwy2 \rightarrow gate2} P_{deb2}$$

$$\lambda_{emb2} = \alpha_{emb2} + \lambda_{deb2} P_{deb2} P_{emb2} + \lambda_{rwy2 \rightarrow gate2} P_{rwy2 \rightarrow gate2} P_{emb2}$$

$$\lambda_{rwy2 \rightarrow gate2} = \lambda_{att2}$$

$$\lambda_{rwy2 \rightarrow rwy2} = \lambda_{emb2} P_{emb2} P_{rwy2 \rightarrow rwy2} + \lambda_{deb2} P_{deb2} P_{rwy2 \rightarrow rwy2}$$

$$\lambda_{dec2} = \lambda_{rwy2 \rightarrow rwy2}$$

$$\lambda_{att2} = \lambda_{ima2 \rightarrow rwy2}$$

$$\lambda_{ima2 \rightarrow rwy2} = \lambda_{sec2 \rightarrow ima2}$$

$$\lambda_{ima2 \rightarrow sec2} = \lambda_{dec2}$$

$$\lambda_{sec2 \rightarrow ima2} = \alpha_{sec2 \rightarrow ima2} + \lambda_{sec1 \rightarrow sec2} P_{sec1 \rightarrow sec2} P_{sec2 \rightarrow ima2} + \lambda_{sec3 \rightarrow sec2} P_{sec3 \rightarrow sec2} P_{sec2 \rightarrow ima2}$$

$$\lambda_{sec2 \rightarrow o} = \lambda_{ima2 \rightarrow sec2} P_{ima2 \rightarrow sec2} P_{sec2 \rightarrow o} + \lambda_{sec1 \rightarrow sec2} P_{sec1 \rightarrow sec2} P_{sec2 \rightarrow o} + \lambda_{sec3 \rightarrow sec2} P_{sec3 \rightarrow sec2} P_{sec2 \rightarrow o}$$

$$\lambda_{sec2 \rightarrow sec1} = \alpha_{sec2 \rightarrow sec1} + \lambda_{ima2 \rightarrow sec2} P_{ima2 \rightarrow sec2} P_{sec2 \rightarrow sec1} + \lambda_{sec3 \rightarrow sec2} P_{sec3 \rightarrow sec2} P_{sec2 \rightarrow sec1}$$

$$\lambda_{sec2 \rightarrow sec3} = \alpha_{sec2 \rightarrow sec3} + \lambda_{ima2 \rightarrow sec2} P_{ima2 \rightarrow sec2} P_{sec2 \rightarrow sec3} + \lambda_{sec1 \rightarrow sec2} P_{sec1 \rightarrow sec2} P_{sec2 \rightarrow sec3}$$

$$\lambda_{deb3} = \lambda_{rwy3 \rightarrow gate3} P_{rwy3 \rightarrow gate3} P_{deb3}$$

$$\lambda_{emb3} = \alpha_{emb3} + \lambda_{deb3} P_{deb3} P_{emb3} + \lambda_{rwy3 \rightarrow gate3} P_{rwy3 \rightarrow gate3} P_{emb3}$$



$$\lambda_{twy3 \rightarrow gate3} = \lambda_{att3}$$

$$\lambda_{twy3 \rightarrow rwy3} = \lambda_{emb3} P_{emb3 \rightarrow rwy3} + \lambda_{deb3 \rightarrow rwy3}$$

$$\lambda_{dec3} = \lambda_{twy3 \rightarrow rwy3}$$

$$\lambda_{att3} = \lambda_{ima3 \rightarrow rwy3}$$

$$\lambda_{ima3 \rightarrow rwy3} = \lambda_{sec3 \rightarrow ima3}$$

$$\lambda_{ima3 \rightarrow sec3} = \lambda_{dec3}$$

$$\lambda_{sec3 \rightarrow ima3} = \alpha_{sec3 \rightarrow ima3} + \lambda_{sec1 \rightarrow sec3} P_{sec1 \rightarrow sec3 \rightarrow ima3} + \lambda_{sec2 \rightarrow sec3} P_{sec2 \rightarrow sec3 \rightarrow ima3}$$

$$\lambda_{sec3 \rightarrow o} = \lambda_{ima3 \rightarrow sec3} P_{ima3 \rightarrow sec3 \rightarrow o} + \lambda_{sec1 \rightarrow sec3} P_{sec1 \rightarrow sec3 \rightarrow o} + \lambda_{sec2 \rightarrow sec3} P_{sec2 \rightarrow sec3 \rightarrow o}$$

$$\lambda_{sec3 \rightarrow sec2} = \alpha_{sec3 \rightarrow sec2} + \lambda_{ima3 \rightarrow sec3} P_{ima3 \rightarrow sec3 \rightarrow sec2} + \lambda_{sec1 \rightarrow sec3} P_{sec1 \rightarrow sec3 \rightarrow sec2}$$

$$\lambda_{sec3 \rightarrow sec1} = \alpha_{sec3 \rightarrow sec1} + \lambda_{ima3 \rightarrow sec3} P_{ima3 \rightarrow sec3 \rightarrow sec1} + \lambda_{sec2 \rightarrow sec3} P_{sec2 \rightarrow sec3 \rightarrow sec1}$$

Nous obtenons, de cette façon, un système de 36 équations à 36 inconnues où les inconnues sont les  $\lambda_j$ . Les probabilités des transitions  $p_{ij}$  et les taux d'arrivée dans le réseau  $\alpha_i$  sont des constantes. Ce système d'équation admet une solution.

### Intensité du trafic nominal des $j$ -avions $\rho_j$

$$\rho_j = \frac{\lambda_j}{\mu_j}, \text{ pour } j \in \mathcal{N}$$

Étant donné  $\mu_j$ , on peut déduire la valeur de  $\rho_j$  à partir de la valeur de  $\lambda_j$  calculée plus haut.

### Intensité du trafic pour les $S$ -avions

$$\rho(S) = \sum_{j \in S} \rho_j \quad S \subseteq \mathcal{N}.$$

### S-charge de travail moyenne d'un $j$ -avion

$$V_i^S = \frac{1}{\mu_i} + \sum_{j \in S} p_{ij} V_j^S, \text{ pour } i \in S, S \subseteq \mathcal{N}.$$

### Intensité du trafic externe

$$\rho^o(S) = \sum_{j \in S} \alpha_j V_j^S$$

$$\begin{aligned}
\rho^o(C_{GATE}) &= \sum_{j \in GATE} \alpha_j V_j^{C_{GATE}} \\
&= \alpha_{emb} V_{emb}^{C_{GATE}} + \alpha_{deb} V_{deb}^{C_{GATE}} \\
&= \alpha_{emb} V_{emb}^{C_{GATE}}
\end{aligned}$$

$$\begin{aligned}
\rho^o(C_{SEC1}) &= \sum_{j \in SEC1} \alpha_j V_j^{C_{SEC1}} \\
&= \alpha_{sec1 \rightarrow o} V_{sec1 \rightarrow o}^{C_{SEC1}} + \alpha_{sec1 \rightarrow sec2} V_{sec1 \rightarrow sec2}^{C_{SEC1}} + \alpha_{sec1 \rightarrow sec3} V_{sec1 \rightarrow sec3}^{C_{SEC1}} + \alpha_{sec1 \rightarrow tma1} V_{sec1 \rightarrow tma1}^{C_{SEC1}} \\
&= \alpha_{sec1 \rightarrow sec2} V_{sec1 \rightarrow sec2}^{C_{SEC1}} + \alpha_{sec1 \rightarrow sec3} V_{sec1 \rightarrow sec3}^{C_{SEC1}} + \alpha_{sec1 \rightarrow tma1} V_{sec1 \rightarrow tma1}^{C_{SEC1}}
\end{aligned}$$

$$\rho^o(C_{TWY}) = \rho^o(C_{RWY}) = \rho^o(C_{TMA}) = 0$$

Si nous prenons  $S = \mathcal{N}$ , alors

$$\rho^o(\mathcal{N}) = \rho^o(C_{GATE1}) + \rho^o(C_{GATE2}) + \rho^o(C_{GATE3}) + \rho^o(C_{SEC1}) + \rho^o(C_{SEC2}) + \rho^o(C_{SEC3})$$

### Pic d'intensité du trafic des *i*-avions dans la station *m*

$$R(i, m) = \mu_i \sum_{j \in C_m} p_{ij} V_j^{C_m}, \text{ pour } i \in \mathcal{N}, m \in \mathcal{M}.$$

Pour calculer ce pic d'intensité du trafic, nous devons considérer chaque station de service de façon séparée. De cette façon, nous pouvons déterminer le pic d'intensité du trafic de chaque classe d'avions servie par cette station.

$$m = GATE$$

$$C_{GATE} = \{emb, deb\}$$

$$\begin{aligned}
R(deb, GATE) &= \mu_{deb} \sum_{j \in C_{GATE}} p_{deb j} V_j^{C_{GATE}} \\
&= \mu_{deb} p_{deb emb} V_{emb}^{C_{GATE}}
\end{aligned}$$

$$\begin{aligned}
V_{emb}^{C_{GATE}} &= \frac{1}{\mu_{emb}} + \sum_{j \in C_{GATE}} p_{emb j} V_j^{C_{GATE}} \\
&= \frac{1}{\mu_{emb}}
\end{aligned}$$

$$R(deb, GATE) = \frac{\mu_{deb}}{\mu_{emb}} p_{deb emb}$$

$$\begin{aligned}
R(twy \rightarrow gate, GATE) &= \mu_{twy \rightarrow gate} \sum_{j \in C_{GATE}} p_{twy \rightarrow gate j} V_j^{C_{GATE}} \\
&= \mu_{twy \rightarrow gate} (p_{twy \rightarrow gate emb} V_{emb}^{C_{GATE}} + p_{twy \rightarrow gate deb} V_{deb}^{C_{GATE}})
\end{aligned}$$



$$\begin{aligned}
V_{deb}^{C_{GATE}} &= \frac{1}{\mu_{deb}} + \sum_{j \in C_{GATE}} p_{deb \ j} V_j^{C_{GATE}} \\
&= \frac{1}{\mu_{deb}} + p_{deb \ emb} V_{emb}^{C_{GATE}} \\
&= \frac{1}{\mu_{deb}} + \frac{1}{\mu_{emb}} p_{deb \ emb}
\end{aligned}$$

$$\begin{aligned}
R(twy \rightarrow gate, GATE) &= \mu_{twy \rightarrow gate} \left( \frac{1}{\mu_{emb}} p_{twy \rightarrow gate \ emb} + p_{twy \rightarrow gate \ deb} \left( \frac{1}{\mu_{deb}} + \frac{1}{\mu_{emb}} p_{deb \ emb} \right) \right) \\
&= \frac{\mu_{twy \rightarrow gate}}{\mu_{emb}} p_{twy \rightarrow gate \ emb} + \frac{\mu_{twy \rightarrow gate}}{\mu_{deb}} p_{twy \rightarrow gate \ deb} + \frac{\mu_{twy \rightarrow gate}}{\mu_{emb}} p_{twy \rightarrow gate \ deb} p_{deb \ emb}
\end{aligned}$$

$$m = TWY$$

$$C_{TWY} = \{twy \rightarrow rwy, twy \rightarrow gate\}$$

$$\begin{aligned}
R(emb, TWY) &= \mu_{emb} \sum_{j \in C_{TWY}} p_{emb \ j} V_j^{C_{TWY}} \\
&= \mu_{emb} p_{emb \ twy \rightarrow rwy} V_{twy \rightarrow rwy}^{C_{TWY}}
\end{aligned}$$

$$\begin{aligned}
V_{twy \rightarrow rwy}^{C_{TWY}} &= \frac{1}{\mu_{twy \rightarrow rwy}} + \sum_{j \in C_{TWY}} p_{twy \rightarrow rwy \ j} V_j^{C_{TWY}} \\
&= \frac{1}{\mu_{twy \rightarrow rwy}}
\end{aligned}$$

$$R(emb, TWY) = \frac{\mu_{emb}}{\mu_{twy \rightarrow rwy}} p_{emb \ twy \rightarrow rwy}$$

$$\begin{aligned}
R(deb, TWY) &= \mu_{deb} \sum_{j \in C_{TWY}} p_{deb \ j} V_j^{C_{TWY}} \\
&= \mu_{deb} p_{deb \ twy \rightarrow rwy} V_{twy \rightarrow rwy}^{C_{TWY}} \\
&= \frac{\mu_{deb}}{\mu_{twy \rightarrow rwy}} p_{deb \ twy \rightarrow rwy}
\end{aligned}$$

$$\begin{aligned}
R(att, TWY) &= \mu_{att} \sum_{j \in C_{TWY}} p_{att \ j} V_j^{C_{TWY}} \\
&= \mu_{att} p_{att \ twy \rightarrow gate} V_{twy \rightarrow gate}^{C_{TWY}}
\end{aligned}$$

$$\begin{aligned}
V_{twy \rightarrow gate}^{C_{TWY}} &= \frac{1}{\mu_{twy \rightarrow gate}} + \sum_{j \in C_{TWY}} p_{twy \rightarrow gate \ j} V_j^{C_{TWY}} \\
&= \frac{1}{\mu_{twy \rightarrow gate}}
\end{aligned}$$

$$R(att, TWY) = \frac{\mu_{att}}{\mu_{twy \rightarrow gate}} p_{att \ twy \rightarrow gate}$$

$$\underline{m = RWY}$$

$$C_{RWY} = \{dec, att\}$$

$$\begin{aligned} R(twy \rightarrow rwy, RWY) &= \mu_{twy \rightarrow rwy} \sum_{j \in C_{RWY}} p_{twy \rightarrow rwy \ j} V_j^{C_{RWY}} \\ &= \mu_{twy \rightarrow rwy} p_{twy \rightarrow rwy \ dec} V_{dec}^{C_{RWY}} \end{aligned}$$

$$\begin{aligned} V_{dec}^{C_{RWY}} &= \frac{1}{\mu_{dec}} + \sum_{j \in C_{RWY}} p_{dec \ j} V_j^{C_{RWY}} \\ &= \frac{1}{\mu_{dec}} \end{aligned}$$

$$R(twy \rightarrow rwy, RWY) = \frac{\mu_{twy \rightarrow rwy}}{\mu_{dec}} p_{twy \rightarrow rwy \ dec}$$

$$\begin{aligned} R(tma \rightarrow rwy, RWY) &= \mu_{tma \rightarrow rwy} \sum_{j \in C_{RWY}} p_{tma \rightarrow rwy \ j} V_j^{C_{RWY}} \\ &= \mu_{tma \rightarrow rwy} p_{tma \rightarrow rwy \ att} V_{att}^{C_{RWY}} \end{aligned}$$

$$\begin{aligned} V_{att}^{C_{RWY}} &= \frac{1}{\mu_{att}} + \sum_{j \in C_{RWY}} p_{att \ j} V_j^{C_{RWY}} \\ &= \frac{1}{\mu_{att}} \end{aligned}$$

$$R(tma \rightarrow rwy, RWY) = \frac{\mu_{tma \rightarrow rwy}}{\mu_{att}} p_{tma \rightarrow rwy \ att}$$

$$\underline{m = TMA}$$

$$C_{TMA} = \{tma \rightarrow rwy, tma \rightarrow sec\}$$

$$\begin{aligned} R(dec, TMA) &= \mu_{dec} \sum_{j \in C_{TMA}} p_{dec \ j} V_j^{C_{TMA}} \\ &= \mu_{dec} p_{dec \ tma \rightarrow sec} V_{tma \rightarrow sec}^{C_{TMA}} \end{aligned}$$

$$\begin{aligned} V_{tma \rightarrow sec}^{C_{TMA}} &= \frac{1}{\mu_{tma \rightarrow sec}} + \sum_{j \in C_{TMA}} p_{tma \rightarrow sec \ j} V_j^{C_{TMA}} \\ &= \frac{1}{\mu_{tma \rightarrow sec}} \end{aligned}$$



$$R(dec, TMA) = \frac{\mu_{dec}}{\mu_{tma \rightarrow sec}} p_{dec \ tma \rightarrow sec}$$

$$\begin{aligned} R(sec \rightarrow tma, TMA) &= \mu_{sec \rightarrow tma} \sum_{j \in C_{TMA}} p_{sec \rightarrow tma \ j} V_j^{C_{TMA}} \\ &= \mu_{sec \rightarrow tma} p_{sec \rightarrow tma \ tma \rightarrow rwy} V_{tma \rightarrow rwy}^{C_{TMA}} \end{aligned}$$

$$\begin{aligned} V_{tma \rightarrow rwy}^{C_{TMA}} &= \frac{1}{\mu_{tma \rightarrow rwy}} + \sum_{j \in C_{TMA}} p_{tma \rightarrow rwy \ j} V_j^{C_{TMA}} \\ &= \frac{1}{\mu_{tma \rightarrow rwy}} \end{aligned}$$

$$R(sec \rightarrow tma, TMA) = \frac{\mu_{sec \rightarrow tma}}{\mu_{tma \rightarrow rwy}} p_{sec \rightarrow tma \ tma \rightarrow rwy}$$

$$m = SEC1$$

$$C_{SEC1} = \{sec1 \rightarrow o, sec1 \rightarrow sec2, sec1 \rightarrow sec3, sec1 \rightarrow tma1\}$$

$$\begin{aligned} R(tma1 \rightarrow sec1, SEC1) &= \mu_{tma1 \rightarrow sec1} \sum_{j \in C_{SEC1}} p_{tma1 \rightarrow sec1 \ j} V_j^{C_{SEC1}} \\ &= \mu_{tma1 \rightarrow sec1} (p_{tma1 \rightarrow sec1 \ sec1 \rightarrow o} V_{sec1 \rightarrow o}^{C_{SEC1}} + \\ &\quad + p_{tma1 \rightarrow sec1 \ sec1 \rightarrow sec2} V_{sec1 \rightarrow sec2}^{C_{SEC1}} + \\ &\quad + p_{tma1 \rightarrow sec1 \ sec1 \rightarrow sec3} V_{sec1 \rightarrow sec3}^{C_{SEC1}}) \end{aligned}$$

$$\begin{aligned} V_{sec1 \rightarrow o}^{C_{SEC1}} &= \frac{1}{\mu_{sec1 \rightarrow o}} + \sum_{j \in C_{SEC1}} p_{sec1 \rightarrow o \ j} V_j^{C_{SEC1}} \\ &= \frac{1}{\mu_{sec1 \rightarrow o}} \end{aligned}$$

$$\begin{aligned} V_{sec1 \rightarrow sec2}^{C_{SEC1}} &= \frac{1}{\mu_{sec1 \rightarrow sec2}} + \sum_{j \in C_{SEC1}} p_{sec1 \rightarrow sec2 \ j} V_j^{C_{SEC1}} \\ &= \frac{1}{\mu_{sec1 \rightarrow sec2}} \end{aligned}$$

$$\begin{aligned} V_{sec1 \rightarrow sec3}^{C_{SEC1}} &= \frac{1}{\mu_{sec1 \rightarrow sec3}} + \sum_{j \in C_{SEC1}} p_{sec1 \rightarrow sec3 \ j} V_j^{C_{SEC1}} \\ &= \frac{1}{\mu_{sec1 \rightarrow sec3}} \end{aligned}$$

$$V_{sec1 \rightarrow tma1}^{C_{SEC1}} = \frac{1}{\mu_{sec1 \rightarrow tma1}} + \sum_{j \in C_{SEC1}} p_{sec1 \rightarrow tma1 j} V_j^{C_{SEC1}}$$

$$= \frac{1}{\mu_{sec1 \rightarrow tma1}}$$

$$R(tma1 \rightarrow sec1, SEC1) = \frac{\mu_{tma1 \rightarrow sec1}}{\mu_{sec1 \rightarrow o}} p_{tma1 \rightarrow sec1 sec1 \rightarrow o} + \frac{\mu_{tma1 \rightarrow sec1}}{\mu_{sec1 \rightarrow sec2}} p_{tma1 \rightarrow sec1 sec1 \rightarrow sec2}$$

$$+ \frac{\mu_{tma1 \rightarrow sec1}}{\mu_{sec1 \rightarrow sec3}} p_{tma1 \rightarrow sec1 sec1 \rightarrow sec3}$$

$$R(sec2 \rightarrow sec1, SEC1) = \mu_{sec2 \rightarrow sec1} \sum_{j \in C_{SEC1}} p_{sec2 \rightarrow sec1 j} V_j^{C_{SEC1}}$$

$$= \frac{\mu_{sec2 \rightarrow sec1}}{\mu_{sec1 \rightarrow o}} p_{sec2 \rightarrow sec1 sec1 \rightarrow o} + \frac{\mu_{sec2 \rightarrow sec1}}{\mu_{sec1 \rightarrow sec3}} p_{sec2 \rightarrow sec1 sec1 \rightarrow sec3}$$

$$+ \frac{\mu_{sec2 \rightarrow sec1}}{\mu_{sec1 \rightarrow tma1}} p_{sec2 \rightarrow sec1 sec1 \rightarrow tma1}$$

avec

$$V_{sec1 \rightarrow tma1}^{C_{SEC1}} = \frac{1}{\mu_{sec1 \rightarrow tma1}} + \sum_{j \in C_{SEC1}} p_{sec1 \rightarrow tma1 j} V_j^{C_{SEC1}}$$

$$= \frac{1}{\mu_{sec1 \rightarrow tma1}}$$

Toutes les autres valeurs de  $R(i, m)$  sont nulles.

Pour l'ensemble du réseau, les autres valeurs de  $R(i, m)$  peuvent être déduites à partir des valeurs calculées ci-dessus. Il suffit d'ajouter les indices correspondants.

### Pic d'intensité du trafic de la station $m'$ vers la station $m$

Par définition,

$$\bar{R}(m', m) = \max_{i \in C_{m'}} R(i, m), \text{ pour } m, m' \in \mathcal{M}, \text{ avec } m' \neq m.$$

Pour le calcul du pic d'intensité du trafic, nous considérons une paire de stations de service qui communiquent en un pas.

$$\bar{R}(GATE, TWY) = \max_{i \in C_{GATE}} R(i, TWY)$$

$$= \max\{R(emb, TWY), R(deb, TWY)\}$$



$$\begin{aligned}\bar{R}(TWY, GATE) &= \max_{i \in C_{TWY}} R(i, GATE) \\ &= R(twy \rightarrow gate, GATE)\end{aligned}$$

$$\begin{aligned}\bar{R}(TWY, RWY) &= \max_{i \in C_{TWY}} R(i, RWY) \\ &= R(twy \rightarrow rwy, RWY)\end{aligned}$$

$$\begin{aligned}\bar{R}(RWY, TWY) &= \max_{i \in C_{RWY}} R(i, TWY) \\ &= R(att, TWY)\end{aligned}$$

$$\begin{aligned}\bar{R}(RWY, TMA) &= \max_{i \in C_{RWY}} R(i, TMA) \\ &= R(dec, TMA)\end{aligned}$$

$$\begin{aligned}\bar{R}(TMA, RWY) &= \max_{i \in C_{TMA}} R(i, RWY) \\ &= R(tma \rightarrow rwy, RWY)\end{aligned}$$

$$\begin{aligned}\bar{R}(TMA, SEC) &= \max_{i \in C_{TMA}} R(i, SEC) \\ &= R(tma \rightarrow sec, SEC)\end{aligned}$$

$$\begin{aligned}\bar{R}(SEC, TMA) &= \max_{i \in C_{SEC}} R(i, TMA) \\ &= R(sec \rightarrow tma, TMA)\end{aligned}$$

$$\begin{aligned}\bar{R}(SEC1, SEC2) &= \max_{i \in C_{SEC1}} R(i, SEC2) \\ &= R(sec1 \rightarrow sec2, SEC2)\end{aligned}$$

**Pic d'intensité du trafic pour la station  $m$**

$$\bar{\rho}(m) = \sum_{j \in C_m} \alpha_j V_j^{C_m} + \sum_{m' \in \mathcal{M} \setminus \{m\}} \bar{R}(m', m), \text{ pour } m \in \mathcal{M}.$$

Puisque le trafic externe passe exclusivement par les *GATES* et les *SECs*, le premier terme de la partie droite de l'équation est non nulle pour ces deux stations de service. Ce terme représente l'intensité du trafic externe.

$$\begin{aligned}\bar{\rho}(GATE) &= \sum_{j \in C_{GATE}} \alpha_j V_j^{C_{GATE}} + \sum_{m' \in \mathcal{M} \setminus \{GATE\}} \bar{R}(m', GATE) \\ &= \alpha_{emb} V_{emb}^{C_{GATE}} + \alpha_{deb} V_{deb}^{C_{GATE}} + \bar{R}(TWY, GATE) \\ &= \frac{\alpha_{emb}}{\mu_{emb}} + \bar{R}(TWY, GATE) \quad \text{car } \alpha_{deb} = 0\end{aligned}$$

$$\bar{\rho}(TWY) = \bar{R}(GATE, TWY) + \bar{R}(RWY, TWY)$$

$$\bar{\rho}(RWY) = \bar{R}(TWY, RWY) + \bar{R}(TMA, RWY)$$

$$\bar{\rho}(TMA) = \bar{R}(RWY, TMA) + \bar{R}(SEC, TMA)$$

$$\begin{aligned} \bar{\rho}(SEC1) = & \frac{\alpha_{sec1 \rightarrow sec2}}{\mu_{sec1 \rightarrow sec2}} + \frac{\alpha_{sec1 \rightarrow sec3}}{\mu_{sec1 \rightarrow sec3}} + \frac{\alpha_{sec1 \rightarrow tma1}}{\mu_{sec1 \rightarrow tma1}} + \\ & + \bar{R}(TMA1, SEC1) + \bar{R}(SEC2, SEC1) + \bar{R}(SEC3, SEC1) \end{aligned}$$

L'état du système à l'instant  $t$  est donné par les variables aléatoires suivantes:

- $L_j(t)$ : nombre total des  $j$ -avions dans le système à l'instant  $t$ ;
- $B_j(t)$ : 1 si un  $j$ -avion est en service au temps  $t$ ; 0 sinon;
- $B^m(t)$ : 1 si le serveur  $m$  est occupé à l'instant  $t$ ; 0 sinon.

Notre analyse recourt fréquemment aux mesures de performance suivantes, définies pour chaque règle admissible:

$$x_j = E[L_j] \quad \text{pour } j \in \mathcal{A}, \quad (4.6)$$

$$x_{ij} = E[L_j | B_i = 1] \quad \text{pour } i, j \in \mathcal{A}, \quad (4.7)$$

et

$$x_j^{0m} = E[L_j | B^m = 0] \quad \text{pour } j \in \mathcal{A}, m \in \mathcal{M} \quad (4.8)$$

où les espérances présentées ci-dessus sont les temps moyens.

## 5.2 Stabilité du modèle MQNET

**Théorème 2** (Condition globale de stabilité et borne de performance) *Si la condition*

$$\rho(C_m) < 1 \text{ et } \bar{\rho}(m) < 1 \text{ pour } m \in \mathcal{M} \quad (4.12)$$

*est satisfaite, alors*

- (iii) *le réseau est stable sous n'importe quelle règle admissible, et la chaîne de Markov qui représente l'évolution du réseau a un moment exponentiel qui converge géométriquement;*



- (iv) sous n'importe quelle règle admissible, le nombre moyen d'avions dans le réseau à l'instant  $t$  est borné comme suit:

$$\sum_{j \in A} E[L_j] \leq \sum_{m \in M} \sum_{j \in C_m} \frac{\rho_j V_j^{C_m}}{(1 - \bar{\rho}(m) V^{C_m})}, \quad (4.13)$$

où

$$V^{C_m} = \min_{j \in C_m} V_j^{C_m}$$

Appliquons ce théorème au cas de la navigation aérienne. Déterminons la relation qui permet de calculer  $\rho(C_m)$  et  $\bar{\rho}(m)$  en fonction des variables du modèle et au niveau de chaque station de service  $m$ .

$$m = \text{GATE}, C_{\text{GATE}} = \{\text{emb}, \text{deb}\}$$

$$\begin{aligned} \rho(C_{\text{GATE}}) &= \sum_{i \in C_{\text{GATE}}} \bar{\rho}_i \\ &= \rho_{\text{emb}} + \rho_{\text{deb}} \\ &= \frac{\lambda_{\text{emb}}}{\mu_{\text{emb}}} + \frac{\lambda_{\text{deb}}}{\mu_{\text{deb}}} \end{aligned}$$

$$\rho(C_{\text{GATE1}}) = \frac{\lambda_{\text{emb1}}}{\mu_{\text{emb1}}} + \frac{\lambda_{\text{deb1}}}{\mu_{\text{deb1}}}$$

$$\rho(C_{\text{GATE2}}) = \frac{\lambda_{\text{emb2}}}{\mu_{\text{emb2}}} + \frac{\lambda_{\text{deb2}}}{\mu_{\text{deb2}}}$$

$$\rho(C_{\text{GATE3}}) = \frac{\lambda_{\text{emb3}}}{\mu_{\text{emb3}}} + \frac{\lambda_{\text{deb3}}}{\mu_{\text{deb3}}}$$

$$\bar{\rho}(\text{GATE1}) = \frac{\alpha_{\text{emb1}}}{\mu_{\text{emb1}}} + \bar{R}(\text{TWY1}, \text{GATE1})$$

$$\bar{\rho}(\text{GATE2}) = \frac{\alpha_{\text{emb2}}}{\mu_{\text{emb2}}} + \bar{R}(\text{TWY2}, \text{GATE2})$$

$$\bar{\rho}(\text{GATE3}) = \frac{\alpha_{\text{emb3}}}{\mu_{\text{emb3}}} + \bar{R}(\text{TWY3}, \text{GATE3})$$

$$m = \text{TWY}, C_{\text{TWY}} = \{\text{twy} \rightarrow \text{gate}, \text{twy} \rightarrow \text{rwy}\}$$

$$\begin{aligned}
\rho(C_{TWY}) &= \sum_{i \in C_{TWY}} \rho_i \\
&= \rho_{twy \rightarrow rwy} + \rho_{twy \rightarrow gate} \\
&= \frac{\lambda_{twy \rightarrow rwy}}{\mu_{twy \rightarrow rwy}} + \frac{\lambda_{twy \rightarrow gate}}{\mu_{twy \rightarrow gate}}
\end{aligned}$$

$$\rho(C_{TWY1}) = \frac{\lambda_{twy1 \rightarrow rwy1}}{\mu_{twy1 \rightarrow rwy1}} + \frac{\lambda_{twy1 \rightarrow gate1}}{\mu_{twy1 \rightarrow gate1}}$$

$$\rho(C_{TWY2}) = \frac{\lambda_{twy2 \rightarrow rwy2}}{\mu_{twy2 \rightarrow rwy2}} + \frac{\lambda_{twy2 \rightarrow gate2}}{\mu_{twy2 \rightarrow gate2}}$$

$$\rho(C_{TWY3}) = \frac{\lambda_{twy3 \rightarrow rwy3}}{\mu_{twy3 \rightarrow rwy3}} + \frac{\lambda_{twy3 \rightarrow gate3}}{\mu_{twy3 \rightarrow gate3}}$$

$$\bar{\rho}(TWY1) = \bar{R}(GATE1, TWY1) + \bar{R}(RWY1, TWY1)$$

$$\bar{\rho}(TWY2) = \bar{R}(GATE2, TWY2) + \bar{R}(RWY2, TWY2)$$

$$\bar{\rho}(TWY3) = \bar{R}(GATE3, TWY3) + \bar{R}(RWY3, TWY3)$$

$$m = RWY, C_{RWY} = \{dec, att\}$$

$$\begin{aligned}
\rho(C_{RWY}) &= \sum_{i \in C_{RWY}} \rho_i \\
&= \rho_{dec} + \rho_{att} \\
&= \frac{\lambda_{dec}}{\mu_{dec}} + \frac{\lambda_{att}}{\mu_{att}}
\end{aligned}$$

$$\rho(C_{RWY1}) = \frac{\lambda_{dec1}}{\mu_{dec1}} + \frac{\lambda_{att1}}{\mu_{att1}}$$

$$\rho(C_{RWY2}) = \frac{\lambda_{dec2}}{\mu_{dec2}} + \frac{\lambda_{att2}}{\mu_{att2}}$$

$$\rho(C_{RWY3}) = \frac{\lambda_{dec3}}{\mu_{dec3}} + \frac{\lambda_{att3}}{\mu_{att3}}$$

$$\bar{\rho}(RWY1) = \bar{R}(TWY1, RWY1) + \bar{R}(TMA1, RWY1)$$

$$\bar{\rho}(RWY2) = \bar{R}(TWY2, RWY2) + \bar{R}(TMA2, RWY2)$$

$$\bar{\rho}(RWY3) = \bar{R}(TWY3, RWY3) + \bar{R}(TMA3, RWY3)$$



$$m = TMA, C_{TMA} = \{tma \rightarrow sec, tma \rightarrow rwy\}$$

$$\begin{aligned} \rho(C_{TMA}) &= \sum_{i \in C_{TMA}} \rho_i \\ &= \rho_{tma \rightarrow rwy} + \rho_{tma \rightarrow gate} \\ &= \frac{\lambda_{tma \rightarrow rwy}}{\mu_{tma \rightarrow rwy}} + \frac{\lambda_{tma \rightarrow gate}}{\mu_{tma \rightarrow gate}} \end{aligned}$$

$$\rho(C_{TMA1}) = \frac{\lambda_{tma1 \rightarrow rwy1}}{\mu_{tma1 \rightarrow rwy1}} + \frac{\lambda_{tma1 \rightarrow gate1}}{\mu_{tma1 \rightarrow gate1}}$$

$$\rho(C_{TMA2}) = \frac{\lambda_{tma2 \rightarrow rwy2}}{\mu_{tma2 \rightarrow rwy2}} + \frac{\lambda_{tma2 \rightarrow gate2}}{\mu_{tma2 \rightarrow gate2}}$$

$$\rho(C_{TMA3}) = \frac{\lambda_{tma3 \rightarrow rwy3}}{\mu_{tma3 \rightarrow rwy3}} + \frac{\lambda_{tma3 \rightarrow gate3}}{\mu_{tma3 \rightarrow gate3}}$$

$$\bar{\rho}(TMA1) = \bar{R}(RWY1, TMA1) + \bar{R}(SEC1, TMA1)$$

$$\bar{\rho}(TMA2) = \bar{R}(RWY2, TMA2) + \bar{R}(SEC2, TMA2)$$

$$\bar{\rho}(TMA2) = \bar{R}(RWY2, TMA2) + \bar{R}(SEC2, TMA2)$$

$$m = SEC1, C_{SEC1} = \{sec1 \rightarrow o, sec1 \rightarrow sec2, sec1 \rightarrow sec3, sec1 \rightarrow tma1\}$$

$$\begin{aligned} \rho(C_{SEC1}) &= \sum_{i \in C_{SEC1}} \rho_i \\ &= \rho_{sec1 \rightarrow o} + \rho_{sec1 \rightarrow sec2} + \rho_{sec1 \rightarrow sec3} + \rho_{sec1 \rightarrow tma1} \\ &= \frac{\lambda_{sec1 \rightarrow o}}{\mu_{sec1 \rightarrow o}} + \frac{\lambda_{sec1 \rightarrow sec2}}{\mu_{sec1 \rightarrow sec2}} + \frac{\lambda_{sec1 \rightarrow sec3}}{\mu_{sec1 \rightarrow sec3}} + \frac{\lambda_{sec1 \rightarrow tma1}}{\mu_{sec1 \rightarrow tma1}} \end{aligned}$$

$$\begin{aligned} \bar{\rho}(SEC1) &= \frac{\alpha_{sec1 \rightarrow sec2}}{\mu_{sec1 \rightarrow sec2}} + \frac{\alpha_{sec1 \rightarrow sec3}}{\mu_{sec1 \rightarrow sec3}} + \frac{\alpha_{sec1 \rightarrow tma1}}{\mu_{sec1 \rightarrow tma1}} + \\ &\quad + \bar{R}(TMA1, SEC1) + \bar{R}(SEC2, SEC1) + \bar{R}(SEC3, SEC1) \end{aligned}$$

$$m = SEC2, C_{SEC2} = \{sec2 \rightarrow o, sec2 \rightarrow sec1, sec2 \rightarrow sec3, sec2 \rightarrow tma2\}$$

$$\rho(C_{SEC2}) = \frac{\lambda_{sec2 \rightarrow o}}{\mu_{sec2 \rightarrow o}} + \frac{\lambda_{sec2 \rightarrow sec1}}{\mu_{sec2 \rightarrow sec1}} + \frac{\lambda_{sec2 \rightarrow sec3}}{\mu_{sec2 \rightarrow sec3}} + \frac{\lambda_{sec2 \rightarrow tma2}}{\mu_{sec2 \rightarrow tma2}}$$

$$\begin{aligned} \bar{\rho}(SEC2) &= \frac{\alpha_{sec2 \rightarrow sec1}}{\mu_{sec2 \rightarrow sec1}} + \frac{\alpha_{sec2 \rightarrow sec3}}{\mu_{sec2 \rightarrow sec3}} + \frac{\alpha_{sec2 \rightarrow tma2}}{\mu_{sec2 \rightarrow tma2}} + \\ &\quad + \bar{R}(TMA2, SEC2) + \bar{R}(SEC1, SEC2) + \bar{R}(SEC3, SEC2) \end{aligned}$$

$$m = SEC3, C_{SEC3} = \{sec3 \rightarrow o, sec3 \rightarrow sec1, sec3 \rightarrow sec2, sec3 \rightarrow tma3\}$$

$$\rho(C_{SEC3}) = \frac{\lambda_{sec3 \rightarrow o}}{\mu_{sec3 \rightarrow o}} + \frac{\lambda_{sec3 \rightarrow sec1}}{\mu_{sec3 \rightarrow sec1}} + \frac{\lambda_{sec3 \rightarrow sec2}}{\mu_{sec3 \rightarrow sec2}} + \frac{\lambda_{sec3 \rightarrow tma3}}{\mu_{sec3 \rightarrow tma3}}$$

$$\begin{aligned} \bar{\rho}(SEC3) = & \frac{\alpha_{sec3 \rightarrow sec1}}{\mu_{sec3 \rightarrow sec1}} + \frac{\alpha_{sec3 \rightarrow sec2}}{\mu_{sec3 \rightarrow sec2}} + \frac{\alpha_{sec3 \rightarrow tma3}}{\mu_{sec3 \rightarrow tma3}} + \\ & + \bar{R}(TMA3, SEC3) + \bar{R}(SEC1, SEC3) + \bar{R}(SEC2, SEC3) \end{aligned}$$

Pour chaque station de service  $m \in \mathcal{M}$ , si  $\rho(C_m) < 1$  et  $\bar{\rho}(m) < 1$ , alors nous avons la relation

$$\sum_{j \in A'} E[L_j] \leq \sum_{m \in \mathcal{M}} \sum_{j \in C_m} \frac{\rho_j V_j^{C_m}}{(1 - \bar{\rho}(m)) V^{C_m}}, \text{ où } V^{C_m} = \min_{j \in C_m} V_j^{C_m}$$

Ceci signifie que le nombre total moyen d'avions dans le réseau est borné. Calculons cette borne supérieure.

$$\text{Déterminons } \sum_{j \in C_m} \frac{\rho_j V_j^{C_m}}{(1 - \bar{\rho}(m)) V^{C_m}}, \quad m \in \mathcal{M}$$

$$m = GATE$$

$$\sum_{j \in C_{GATE1}} \frac{\rho_j V_j^{C_{GATE1}}}{(1 - \bar{\rho}(GATE1)) V^{C_{GATE1}}} = \frac{\rho_{emb1} V_{emb1}^{C_{GATE1}}}{(1 - \bar{\rho}(GATE1)) V^{C_{GATE1}}} + \frac{\rho_{deb1} V_{deb1}^{C_{GATE1}}}{(1 - \bar{\rho}(GATE1)) V^{C_{GATE1}}}$$

$$\text{où } V^{C_{GATE1}} = \min\{V_{emb1}^{C_{GATE1}}, V_{deb1}^{C_{GATE1}}\}$$

$$\sum_{j \in C_{GATE2}} \frac{\rho_j V_j^{C_{GATE2}}}{(1 - \bar{\rho}(GATE2)) V^{C_{GATE2}}} = \frac{\rho_{emb2} V_{emb2}^{C_{GATE2}}}{(1 - \bar{\rho}(GATE2)) V^{C_{GATE2}}} + \frac{\rho_{deb2} V_{deb2}^{C_{GATE2}}}{(1 - \bar{\rho}(GATE2)) V^{C_{GATE2}}}$$

$$\text{où } V^{C_{GATE2}} = \min\{V_{emb2}^{C_{GATE2}}, V_{deb2}^{C_{GATE2}}\}$$

$$\sum_{j \in C_{GATE3}} \frac{\rho_j V_j^{C_{GATE3}}}{(1 - \bar{\rho}(GATE3)) V^{C_{GATE3}}} = \frac{\rho_{emb3} V_{emb3}^{C_{GATE3}}}{(1 - \bar{\rho}(GATE3)) V^{C_{GATE3}}} + \frac{\rho_{deb3} V_{deb3}^{C_{GATE3}}}{(1 - \bar{\rho}(GATE3)) V^{C_{GATE3}}}$$

$$\text{où } V^{C_{GATE3}} = \min\{V_{emb3}^{C_{GATE3}}, V_{deb3}^{C_{GATE3}}\}$$



$m = TWY$

$$\sum_{j \in C_{TWY1}} \frac{\rho_j V_j^{C_{TWY1}}}{(1 - \bar{\rho}(TWY1)) V^{C_{TWY1}}} = \frac{\rho_{twy1 \rightarrow gate1} V_{twy1 \rightarrow gate1}^{C_{TWY1}}}{(1 - \bar{\rho}(TWY1)) V^{C_{TWY1}}} + \frac{\rho_{twy1 \rightarrow rwy1} V_{twy1 \rightarrow rwy1}^{C_{TWY1}}}{(1 - \bar{\rho}(TWY1)) V^{C_{TWY1}}}$$

$$\text{où } V^{C_{TWY1}} = \min \{ V_{twy1 \rightarrow gate1}^{C_{TWY1}}, V_{twy1 \rightarrow rwy1}^{C_{TWY1}} \}$$

$$\sum_{j \in C_{TWY2}} \frac{\rho_j V_j^{C_{TWY2}}}{(1 - \bar{\rho}(TWY2)) V^{C_{TWY2}}} = \frac{\rho_{twy2 \rightarrow gate2} V_{twy2 \rightarrow gate2}^{C_{TWY2}}}{(1 - \bar{\rho}(TWY2)) V^{C_{TWY2}}} + \frac{\rho_{twy2 \rightarrow rwy2} V_{twy2 \rightarrow rwy2}^{C_{TWY2}}}{(1 - \bar{\rho}(TWY2)) V^{C_{TWY2}}}$$

$$\text{où } V^{C_{TWY2}} = \min \{ V_{twy2 \rightarrow gate2}^{C_{TWY2}}, V_{twy2 \rightarrow rwy2}^{C_{TWY2}} \}$$

$$\sum_{j \in C_{TWY3}} \frac{\rho_j V_j^{C_{TWY3}}}{(1 - \bar{\rho}(TWY3)) V^{C_{TWY3}}} = \frac{\rho_{twy3 \rightarrow gate3} V_{twy3 \rightarrow gate3}^{C_{TWY3}}}{(1 - \bar{\rho}(TWY3)) V^{C_{TWY3}}} + \frac{\rho_{twy3 \rightarrow rwy3} V_{twy3 \rightarrow rwy3}^{C_{TWY3}}}{(1 - \bar{\rho}(TWY3)) V^{C_{TWY3}}}$$

$$\text{où } V^{C_{TWY3}} = \min \{ V_{twy3 \rightarrow gate3}^{C_{TWY3}}, V_{twy3 \rightarrow rwy3}^{C_{TWY3}} \}$$

$m = Rwy$

$$\sum_{j \in C_{RWY1}} \frac{\rho_j V_j^{C_{RWY1}}}{(1 - \bar{\rho}(RWY1)) V^{C_{RWY1}}} = \frac{\rho_{dec1} V_{dec1}^{C_{RWY1}}}{(1 - \bar{\rho}(RWY1)) V^{C_{RWY1}}} + \frac{\rho_{att1} V_{att1}^{C_{RWY1}}}{(1 - \bar{\rho}(RWY1)) V^{C_{RWY1}}}$$

$$\text{où } V^{C_{RWY1}} = \min \{ V_{dec1}^{C_{RWY1}}, V_{att1}^{C_{RWY1}} \}$$

$$\sum_{j \in C_{RWY2}} \frac{\rho_j V_j^{C_{RWY2}}}{(1 - \bar{\rho}(RWY2)) V^{C_{RWY2}}} = \frac{\rho_{dec2} V_{dec2}^{C_{RWY2}}}{(1 - \bar{\rho}(RWY2)) V^{C_{RWY2}}} + \frac{\rho_{att2} V_{att2}^{C_{RWY2}}}{(1 - \bar{\rho}(RWY2)) V^{C_{RWY2}}}$$

$$\text{où } V^{C_{RWY2}} = \min \{ V_{dec2}^{C_{RWY2}}, V_{att2}^{C_{RWY2}} \}$$

$$\sum_{j \in C_{RWY3}} \frac{\rho_j V_j^{C_{RWY3}}}{(1 - \bar{\rho}(RWY3)) V^{C_{RWY3}}} = \frac{\rho_{dec3} V_{dec3}^{C_{RWY3}}}{(1 - \bar{\rho}(RWY3)) V^{C_{RWY3}}} + \frac{\rho_{att3} V_{att3}^{C_{RWY3}}}{(1 - \bar{\rho}(RWY3)) V^{C_{RWY3}}}$$

$$\text{où } V^{C_{RWY3}} = \min \{ V_{dec3}^{C_{RWY3}}, V_{att3}^{C_{RWY3}} \}$$

$m = TMA$

$$\sum_{j \in C_{TMA1}} \frac{\rho_j V_j^{C_{TMA1}}}{(1 - \bar{\rho}(TMA1)) V^{C_{TMA1}}} = \frac{\rho_{tma1 \rightarrow rwy1} V_{tma1 \rightarrow rwy1}^{C_{TMA1}}}{(1 - \bar{\rho}(TMA1)) V^{C_{TMA1}}} + \frac{\rho_{tma1 \rightarrow sec1} V_{tma1 \rightarrow sec1}^{C_{TMA1}}}{(1 - \bar{\rho}(TMA1)) V^{C_{TMA1}}}$$

$$\text{où } V^{C_{TMA1}} = \min \left\{ V_{tma1 \rightarrow rwy1}^{C_{TMA1}}, V_{tma1 \rightarrow sec1}^{C_{TMA1}} \right\}$$

$$\sum_{j \in C_{TMA2}} \frac{\rho_j V_j^{C_{TMA2}}}{(1 - \bar{\rho}(TMA2)) V^{C_{TMA2}}} = \frac{\rho_{tma2 \rightarrow rwy2} V_{tma2 \rightarrow rwy2}^{C_{TMA2}}}{(1 - \bar{\rho}(TMA2)) V^{C_{TMA2}}} + \frac{\rho_{tma2 \rightarrow sec2} V_{tma2 \rightarrow sec2}^{C_{TMA2}}}{(1 - \bar{\rho}(TMA2)) V^{C_{TMA2}}}$$

$$\text{où } V^{C_{TMA2}} = \min \left\{ V_{tma2 \rightarrow rwy2}^{C_{TMA2}}, V_{tma2 \rightarrow sec2}^{C_{TMA2}} \right\}$$

$$\sum_{j \in C_{TMA3}} \frac{\rho_j V_j^{C_{TMA3}}}{(1 - \bar{\rho}(TMA3)) V^{C_{TMA3}}} = \frac{\rho_{tma3 \rightarrow rwy3} V_{tma3 \rightarrow rwy3}^{C_{TMA3}}}{(1 - \bar{\rho}(TMA3)) V^{C_{TMA3}}} + \frac{\rho_{tma3 \rightarrow sec3} V_{tma3 \rightarrow sec3}^{C_{TMA3}}}{(1 - \bar{\rho}(TMA3)) V^{C_{TMA3}}}$$

$$\text{où } V^{C_{TMA3}} = \min \left\{ V_{tma3 \rightarrow rwy3}^{C_{TMA3}}, V_{tma3 \rightarrow sec3}^{C_{TMA3}} \right\}$$

$m = SEC$

$$\begin{aligned} \sum_{j \in C_{SEC1}} \frac{\rho_j V_j^{C_{SEC1}}}{(1 - \bar{\rho}(SEC1)) V^{C_{SEC1}}} &= \frac{\rho_{sec1 \rightarrow o} V_{sec1 \rightarrow o}^{C_{SEC1}}}{(1 - \bar{\rho}(SEC1)) V^{C_{SEC1}}} + \frac{\rho_{sec1 \rightarrow sec2} V_{sec1 \rightarrow sec2}^{C_{SEC1}}}{(1 - \bar{\rho}(SEC1)) V^{C_{SEC1}}} + \frac{\rho_{sec1 \rightarrow sec3} V_{sec1 \rightarrow sec3}^{C_{SEC1}}}{(1 - \bar{\rho}(SEC1)) V^{C_{SEC1}}} + \\ &+ \frac{\rho_{sec1 \rightarrow tma1} V_{sec1 \rightarrow tma1}^{C_{SEC1}}}{(1 - \bar{\rho}(SEC1)) V^{C_{SEC1}}} \end{aligned}$$

$$\text{où } V^{C_{SEC1}} = \min \left\{ V_{sec1 \rightarrow o}^{C_{SEC1}}, V_{sec1 \rightarrow sec2}^{C_{SEC1}}, V_{sec1 \rightarrow sec3}^{C_{SEC1}}, V_{sec1 \rightarrow tma1}^{C_{SEC1}} \right\}$$

$$\begin{aligned} \sum_{j \in C_{SEC2}} \frac{\rho_j V_j^{C_{SEC2}}}{(1 - \bar{\rho}(SEC2)) V^{C_{SEC2}}} &= \frac{\rho_{sec2 \rightarrow o} V_{sec2 \rightarrow o}^{C_{SEC2}}}{(1 - \bar{\rho}(SEC2)) V^{C_{SEC2}}} + \frac{\rho_{sec2 \rightarrow sec1} V_{sec2 \rightarrow sec1}^{C_{SEC2}}}{(1 - \bar{\rho}(SEC2)) V^{C_{SEC2}}} + \frac{\rho_{sec2 \rightarrow sec3} V_{sec2 \rightarrow sec3}^{C_{SEC2}}}{(1 - \bar{\rho}(SEC2)) V^{C_{SEC2}}} + \\ &+ \frac{\rho_{sec2 \rightarrow tma2} V_{sec2 \rightarrow tma2}^{C_{SEC2}}}{(1 - \bar{\rho}(SEC2)) V^{C_{SEC2}}} \end{aligned}$$

$$\text{où } V^{C_{SEC2}} = \min \left\{ V_{sec2 \rightarrow o}^{C_{SEC2}}, V_{sec2 \rightarrow sec1}^{C_{SEC2}}, V_{sec2 \rightarrow sec3}^{C_{SEC2}}, V_{sec2 \rightarrow tma2}^{C_{SEC2}} \right\}$$



$$\sum_{j \in C_{SEC3}} \frac{\rho_j V_j^{C_{SEC3}}}{(1 - \bar{\rho}(SEC3)) V^{C_{SEC3}}} = \frac{\rho_{sec3 \rightarrow o} V_{sec3 \rightarrow o}^{C_{SEC3}}}{(1 - \bar{\rho}(SEC3)) V^{C_{SEC3}}} + \frac{\rho_{sec3 \rightarrow sec1} V_{sec3 \rightarrow sec1}^{C_{SEC3}}}{(1 - \bar{\rho}(SEC3)) V^{C_{SEC3}}} + \frac{\rho_{sec3 \rightarrow sec2} V_{sec3 \rightarrow sec2}^{C_{SEC3}}}{(1 - \bar{\rho}(SEC3)) V^{C_{SEC3}}} + \frac{\rho_{sec3 \rightarrow tma3} V_{sec3 \rightarrow tma3}^{C_{SEC3}}}{(1 - \bar{\rho}(SEC3)) V^{C_{SEC3}}}$$

$$\text{où } V^{C_{SEC1}} = \min \{ V_{sec3 \rightarrow o}^{C_{SEC3}}, V_{sec3 \rightarrow sec1}^{C_{SEC3}}, V_{sec3 \rightarrow sec2}^{C_{SEC3}}, V_{sec3 \rightarrow tma3}^{C_{SEC3}} \}$$

La borne supérieure du nombre total d'avions dans le réseau est obtenue en additionnant les

valeurs des  $\sum_{j \in C_m} \frac{\rho_j V_j^{C_m}}{(1 - \bar{\rho}(m)) V^{C_m}}, \quad m \in \mathcal{M}$  calculées ci-dessus.

### 5.3 L'optimisation approximative des règles d'index par les lois de conservation approximatives

Le problème d'ordonnancement optimal consiste d'une part à trouver une règle d'ordonnancement qui minimise  $c_1 E[L_1] + \dots + c_N E[L_N]$ , et d'autre part à évaluer le coût minimal correspondant  $Z^{OPT}$ .

La règle d'ordonnancement optimale est donnée par une règle index-priorité. Le vecteur d'indices de priorité optimal est calculé par l'algorithme de Klimov que nous avons présenté au chapitre précédent.

#### Algorithme de Klimov adapté

Arguments de l'algorithme:  $\mathbf{c}$  et  $\mathbf{A}$

$$\mathbf{c} = (c_j)_{j \in \mathcal{A}}$$

$\mathbf{c}$  représente le vecteur des coûts encourus par les différents  $j$ -avions.

$$\mathbf{A} = (A_j^S)_{j \in S, S \subseteq \mathcal{A}}$$

$$A_i^S = \begin{cases} V_i^{S \cap C_m} & \text{si } i \in S \cap C_m \\ 0 & \text{sinon} \end{cases}$$

Quel que soit l'ensemble  $S \subseteq \mathcal{N}$ , nous avons soit  $S \cap C_m = \emptyset$  soit  $S \cap C_m \subseteq C_m$

$A$  est une matrice dont les lignes constituent les différentes classes d'avions du réseau et dont les colonnes représentent les  $C_m$ .

Calculons les éléments de la matrice  $A$  pour un ensemble  $S \subseteq \mathcal{N}$  quelconque.

Prenons comme exemple  $m = GATE$ .  $C_m = \{emb, deb\}$

Voici les différents cas qu'il est possible d'obtenir à partir de  $S \cap C_m$ :

- $S \cap C_m = \emptyset$ ; dans ce cas,  $A_i^S = 0$
- $S \cap C_m = \{emb, deb\} = C_{GATE}$  dans ce cas,  $A_i^S = V_i^{C_{GATE}}$   $i \in C_{GATE}$
- $S \cap C_m = \{emb\}$  dans ce cas,  $A_i^S = V_{emb}^{\{emb\}} = V_i^{C_{GATE}}$   $i \in C_{GATE}$
- $S \cap C_m = \{deb\}$  dans ce cas,  $A_i^S = V_{deb}^{\{deb\}}$

Ce calcul peut être fait aisément pour toutes les stations de service du réseau.

Résultats de l'algorithme:  $\pi, \bar{y}, \gamma$

$\pi = (\pi_1, \dots, \pi_N)$ . Ce sont des permutations de  $\mathcal{N}$ .

$\bar{y} = (\bar{y}(S))_{S \subseteq \mathcal{N}}$

$\gamma = (\gamma_1, \dots, \gamma_N)$

Les détails de l'algorithme sont présentés au chapitre précédent.

Supposons que le vecteur de performance  $\mathbf{x}$  satisfait aux lois de conservation approximatives.

Exprimons l'expression  $c_I E[L_I] + \dots + c_N E[L_N]$  comme  $c_I x_I + \dots + c_N x_N$ .

La solution heuristique de  $Z^{OPT} = \min\{c_I x_I + \dots + c_N x_N : \mathbf{x} \in \mathcal{X}\}$  est le vecteur de performance  $\gamma$  obtenu par l'algorithme de Klimov. Cette valeur est notée  $Z^{INDEX}$ .



La solution proposée pour le dual  $Z^{LP} = \min\{c_1 x_1 + \dots + c_N x_N : \mathbf{x} \in \mathcal{P}\}$  est donné par le vecteur  $\bar{\mathbf{y}}$  obtenu également par l'algorithme de Klimov. Cette valeur atteinte est  $Z^D$ . Une des solutions est

$$Z^D = \gamma_1 b(\{1, \dots, N\}) + (\gamma_2 - \gamma_1) b(\{2, \dots, N\}) + \dots + (\gamma_N - \gamma_{N-1}) b(\{N\}) \quad (4.31)$$

Notons que  $Z^D \leq Z^{LP} \leq Z^{OPT} \leq Z^{INDEX}$

**Théorème 4** (Garantie de performance) *Supposons que le vecteur de performance  $\mathbf{x}$  satisfait les lois de conservation approximatives (26) et (27) avec les paramètres  $\mathbf{A}$ ,  $\mathbf{b}$  et  $\Phi$ . De ce fait,*

$$(c) \quad Z^{INDEX} \leq Z^{OPT} + \varepsilon \quad (4.33)$$

et

$$Z^D \geq Z^{OPT} - \varepsilon \quad (4.34)$$

$$\text{où } \varepsilon = \gamma_1 \Phi(\{1, \dots, N\}) + (\gamma_2 - \gamma_1) \Phi(\{2, \dots, N\}) + \dots + (\gamma_N - \gamma_{N-1}) \Phi(\{N\}) \quad (4.35)$$

$$(d) \quad \text{En outre } Z^{INDEX} \leq r Z^{OPT} \quad (4.36)$$

$$\text{et } Z^D \geq \frac{1}{r} Z^{OPT} \quad (4.37)$$

$$\text{où } r = 1 + \max_{S \subseteq \mathcal{A}} \frac{\Phi(S)}{b(S)} \quad (4.38)$$

Pour appliquer le Théorème 4, il faut recourir aux résultats du Théorème 5. Nous devons également calculer le  $(m, m')$ -pic d'intensité du trafic de la classe  $i \in C_m$  vers  $S \cap C_m$  pour  $S \subseteq \mathcal{A}$ ,  $m \in \mu(S)$ ,  $m \neq m'$ .

Par définition,  $R_{m',m}(i, S \cap C_m) = \mu_i \sum_{j \in S \cap C_m} p_{ij} V_j^{S \cap C_m} \quad i \in C_{m'}$

$m' = GATE$  et  $m = TWY$

$$R_{GATE, TWY}(i, S \cap C_{TWY}) = \mu_i \sum_{j \in S \cap C_{TWY}} p_{ij} V_j^{S \cap C_{TWY}} \quad i \in C_{GATE}$$

Si  $S \cap C_{TWY} = C_{TWY}$ , alors  $R_{GATE, TWY}(i, C_{TWY}) = \mu_i \sum_{j \in C_{TWY}} p_{ij} V_j^{C_{TWY}} \quad i \in C_{GATE}$

$$\begin{aligned} R_{GATE, TWY}(deb, C_{TWY}) &= \mu_{deb} \sum_{j \in C_{TWY}} p_{deb j} V_j^{C_{TWY}} \\ &= \frac{\mu_{deb}}{\mu_{twy \rightarrow rwy}} P_{deb twy \rightarrow rwy} \end{aligned}$$

$$\begin{aligned} R_{GATE, TWY}(emb, C_{TWY}) &= \mu_{emb} \sum_{j \in C_{TWY}} p_{emb j} V_j^{C_{TWY}} \\ &= \frac{\mu_{emb}}{\mu_{twy \rightarrow rwy}} P_{emb twy \rightarrow rwy} \end{aligned}$$

Si  $S \cap C_{TWY} = \{twy \rightarrow rwy\}$ , alors  $R_{GATE, TWY}(i, \{twy \rightarrow rwy\}) = \mu_i p_{i twy \rightarrow rwy} V_{twy \rightarrow rwy}^{\{twy \rightarrow rwy\}}$

$$\begin{aligned} R_{GATE, TWY}(deb, \{twy \rightarrow rwy\}) &= \mu_{deb} p_{deb twy \rightarrow rwy} V_{twy \rightarrow rwy}^{\{twy \rightarrow rwy\}} \\ &= \frac{\mu_{deb}}{\mu_{twy \rightarrow rwy}} P_{deb twy \rightarrow rwy} \end{aligned}$$

$$\begin{aligned} R_{GATE, TWY}(emb, \{twy \rightarrow rwy\}) &= \mu_{emb} p_{emb twy \rightarrow rwy} V_{twy \rightarrow rwy}^{\{twy \rightarrow rwy\}} \\ &= \frac{\mu_{emb}}{\mu_{twy \rightarrow rwy}} P_{emb twy \rightarrow rwy} \end{aligned}$$

Si  $S \cap C_{TWY} = \{twy \rightarrow gate\}$ , alors  $R_{GATE, TWY}(i, \{twy \rightarrow gate\}) = \mu_i p_{i twy \rightarrow gate} V_{twy \rightarrow gate}^{\{twy \rightarrow gate\}}$

$$R_{GATE, TWY}(deb, \{twy \rightarrow gate\}) = \mu_{deb} p_{deb twy \rightarrow gate} V_{twy \rightarrow gate}^{\{twy \rightarrow gate\}} = 0$$

$$R_{GATE, TWY}(emb, \{twy \rightarrow gate\}) = \mu_{emb} p_{emb twy \rightarrow gate} V_{twy \rightarrow gate}^{\{twy \rightarrow gate\}} = 0$$

Si  $S \cap C_{TWY} = \emptyset$ , alors  $R_{GATE, TWY}(i, \emptyset) = 0$



$m' = TWY$  et  $m \doteq GATE$

$$R_{TWY, GATE}(i, S \cap C_{GATE}) = \mu_i \sum_{j \in S \cap C_{GATE}} p_{ij} V_j^{S \cap C_{GATE}} \quad i \in C_{TWY}$$

Si  $S \cap C_{GATE} = C_{GATE}$ , alors  $R_{TWY, GATE}(i, C_{GATE}) = \mu_i \sum_{j \in C_{GATE}} p_{ij} V_j^{C_{GATE}} \quad i \in C_{TWY}$

$$\begin{aligned} R_{TWY, GATE}(twy \rightarrow gate, C_{GATE}) &= \mu_{twy \rightarrow gate} \sum_{j \in C_{GATE}} p_{twy \rightarrow gate j} V_j^{C_{GATE}} \\ &= \frac{\mu_{twy \rightarrow gate}}{\mu_{emb}} p_{twy \rightarrow gate emb} + \mu_{twy \rightarrow gate} p_{twy \rightarrow gate deb} \left( \frac{1}{\mu_{deb}} + \frac{1}{\mu_{emb}} p_{deb emb} \right) \end{aligned}$$

$$R_{TWY, GATE}(twy \rightarrow rwy, C_{GATE}) = \mu_{twy \rightarrow rwy} \sum_{j \in C_{GATE}} p_{twy \rightarrow rwy j} V_j^{C_{GATE}} = 0$$

Si  $S \cap C_{GATE} = \{deb\}$ , alors  $R_{TWY, GATE}(i, \{deb\}) = \mu_i p_{i deb} V_{deb}^{\{deb\}}$

$$\begin{aligned} R_{TWY, GATE}(twy \rightarrow gate, \{deb\}) &= \mu_{twy \rightarrow gate} p_{twy \rightarrow gate deb} V_{deb}^{\{deb\}} \\ &= \frac{\mu_{twy \rightarrow gate}}{\mu_{deb}} p_{twy \rightarrow gate deb} \end{aligned}$$

$$R_{TWY, GATE}(twy \rightarrow rwy, \{deb\}) = \mu_{twy \rightarrow rwy} p_{twy \rightarrow rwy deb} V_{deb}^{\{deb\}} = 0$$

Si  $S \cap C_{GATE} = \{emb\}$ , alors  $R_{TWY, GATE}(i, \{emb\}) = \mu_i p_{i emb} V_{emb}^{\{emb\}}$

$$\begin{aligned} R_{TWY, GATE}(twy \rightarrow gate, \{emb\}) &= \mu_{twy \rightarrow gate} p_{twy \rightarrow gate emb} V_{emb}^{\{emb\}} \\ &= \frac{\mu_{twy \rightarrow gate}}{\mu_{emb}} p_{twy \rightarrow gate emb} \end{aligned}$$

$$R_{TWY, GATE}(twy \rightarrow rwy, \{emb\}) = \mu_{twy \rightarrow rwy} p_{twy \rightarrow rwy emb} V_{emb}^{\{emb\}} = 0$$

Si  $S \cap C_{GATE} = \emptyset$ , alors  $R_{TWY, GATE}(i, \emptyset) = 0$

À partir du résultat obtenu pour  $S \cap C_{TWY} = C_{TWY}$ , nous pouvons déduire le résultat des autres cas en tenant compte des transitions élémentaires possibles. C'est pourquoi par la suite, nous donnerons seulement la relation pour  $S \cap C_m = C_m$

**$m' = TWY$  et  $m = RWY$**

$$S \cap C_{RWY} = C_{RWY}$$

$$R_{TWY, RWY}(twy \rightarrow rwy, C_{RWY}) = \frac{\mu_{twy \rightarrow rwy}}{\mu_{dec}} p_{twy \rightarrow rwy dec}$$

$$R_{TWY, RWY}(twy \rightarrow gate, C_{RWY}) = 0$$

$$S \cap C_{RWY} = \{att\}$$

$$R_{TWY, RWY}(twy \rightarrow rwy, \{att\}) = 0$$

$$R_{TWY, RWY}(twy \rightarrow gate, \{att\}) = 0$$

$$S \cap C_{RWY} = \{dec\}$$

$$R_{TWY, RWY}(twy \rightarrow rwy, \{dec\}) = \frac{\mu_{twy \rightarrow rwy}}{\mu_{dec}} p_{twy \rightarrow rwy dec}$$

$$R_{TWY, RWY}(twy \rightarrow gate, \{dec\}) = 0$$

$$S \cap C_{GATE} = \emptyset$$

$$R_{TWY, RWY}(i, \emptyset) = 0$$

**$m' = RWY$  et  $m = TWY$**

$$S \cap C_{TWY} = C_{TWY}$$



$$R_{RWY, TWY}(att, C_{TWY}) = \frac{\mu_{att}}{\mu_{twy \rightarrow gate}} P_{att \ twy \rightarrow gate}$$

$$R_{RWY, TWY}(dec, C_{TWY}) = 0$$

$$S \cap C_{TWY} = \{twy \rightarrow gate\}$$

$$R_{RWY, TWY}(att, \{twy \rightarrow gate\}) = \frac{\mu_{att}}{\mu_{twy \rightarrow gate}} P_{att \ twy \rightarrow gate}$$

$$R_{RWY, TWY}(dec, \{twy \rightarrow gate\}) = 0$$

$$S \cap C_{TWY} = \{twy \rightarrow rwy\}$$

$$R_{RWY, TWY}(dec, \{twy \rightarrow rwy\}) = 0$$

$$R_{RWY, TWY}(att, \{twy \rightarrow rwy\}) = 0$$

$$S \cap C_{TWY} = \emptyset$$

$$R_{RWY, TWY}(i, \emptyset) = 0$$

$$m' = TMA \text{ et } m = RWY$$

$$S \cap C_{RWY} = C_{RWY}$$

$$R_{TMA, RWY}(tma \rightarrow rwy, C_{RWY}) = \frac{\mu_{tma \rightarrow rwy}}{\mu_{att}} P_{tma \rightarrow rwy \ att}$$

$$R_{TMA, RWY}(tma \rightarrow sec, C_{RWY}) = 0$$

$$S \cap C_{RWY} = \{att\}$$

$$R_{TMA,RWY}(tma \rightarrow rwy, \{att\}) = \frac{\mu_{tma \rightarrow rwy}}{\mu_{att}} p_{tma \rightarrow rwy att}$$

$$R_{TMA,RWY}(twy \rightarrow sec, \{att\}) = 0$$

$$S \cap C_{RWY} = \{dec\}$$

$$R_{TMA,RWY}(tma \rightarrow rwy, \{dec\}) = 0$$

$$R_{TMA,RWY}(tma \rightarrow sec, \{dec\}) = 0$$

$$S \cap C_{GATE} = \emptyset$$

$$R_{TMA,RWY}(i, \emptyset) = 0$$

$$m' = RWY \text{ et } m = TMA$$

$$S \cap C_{TMA} = C_{TMA}$$

$$R_{RWY,TMA}(dec, C_{TMA}) = \frac{\mu_{dec}}{\mu_{tma \rightarrow sec}} p_{dec tma \rightarrow sec}$$

$$R_{RWY,TMA}(att, C_{TMA}) = 0$$

$$S \cap C_{TMA} = \{tma \rightarrow sec\}$$

$$R_{RWY,TMA}(dec, \{tma \rightarrow sec\}) = \frac{\mu_{dec}}{\mu_{tma \rightarrow sec}} p_{dec tma \rightarrow sec}$$

$$R_{RWY,TMA}(att, \{tma \rightarrow sec\}) = 0$$

$$S \cap C_{TMA} = \{tma \rightarrow rwy\}$$

$$R_{RWY,TMA}(dec, \{tma \rightarrow rwy\}) = 0$$



$$R_{RWY,TMA}(\text{att}, \{tma \rightarrow rwy\}) = 0$$

$$S \cap C_{TMA} = \emptyset$$

$$R_{RWY,TMA}(i, \emptyset) = 0$$

$$m' = TMA1 \text{ et } m = SEC1$$

$$S \cap C_{SEC1} = \{sec1 \rightarrow o, sec1 \rightarrow sec2, sec1 \rightarrow sec3, sec1 \rightarrow tma1\}$$

$$R_{TMA1,SEC1}(tma1 \rightarrow rwy1, S \cap C_{SEC1}) = 0$$

$$\begin{aligned} R_{TMA1,SEC1}(tma1 \rightarrow sec1, S \cap C_{SEC1}) &= \frac{\mu_{tma1 \rightarrow sec1}}{\mu_{sec1 \rightarrow o}} p_{tma1 \rightarrow sec1 \ sec1 \rightarrow o} \\ &+ \frac{\mu_{tma1 \rightarrow sec1}}{\mu_{sec1 \rightarrow sec2}} p_{tma1 \rightarrow sec1 \ sec1 \rightarrow sec2} \\ &+ \frac{\mu_{tma1 \rightarrow sec1}}{\mu_{sec1 \rightarrow sec3}} p_{tma1 \rightarrow sec1 \ sec1 \rightarrow sec3} \end{aligned}$$

$$m' = SEC1 \text{ et } m = TMA1$$

$$S \cap C_{TMA1} = C_{TMA1} = \{tma1 \rightarrow sec1, tma1 \rightarrow rwy1\}$$

$$R_{SEC1,TMA1}(sec1 \rightarrow o, S \cap C_{TMA1}) = 0$$

$$R_{SEC1,TMA1}(sec1 \rightarrow sec2, S \cap C_{TMA1}) = 0$$

$$R_{SEC1,TMA1}(sec1 \rightarrow sec3, S \cap C_{TMA1}) = 0$$

$$R_{SEC1,TMA1}(sec1 \rightarrow tma1, S \cap C_{TMA1}) = \frac{\mu_{sec1 \rightarrow tma1}}{\mu_{tma1 \rightarrow rwy1}} p_{sec1 \rightarrow tma1 \ tma1 \rightarrow rwy1}$$

$$m' = SEC2 \text{ et } m = SEC1$$

$$S \cap C_{SEC1} = \{sec1 \rightarrow o, sec1 \rightarrow sec2, sec1 \rightarrow sec3, sec1 \rightarrow tma1\}$$

$$R_{SEC2,SEC1}(sec2 \rightarrow o, S \cap C_{SEC1}) = 0$$

$$\begin{aligned} R_{SEC2,SEC1}(sec2 \rightarrow sec1, S \cap C_{TMA1}) &= \frac{\mu_{sec2 \rightarrow sec1}}{\mu_{sec1 \rightarrow o}} p_{sec2 \rightarrow sec1 \ sec1 \rightarrow o} \\ &+ \frac{\mu_{sec2 \rightarrow sec1}}{\mu_{sec1 \rightarrow sec3}} p_{sec2 \rightarrow sec1 \ sec1 \rightarrow sec3} \\ &+ \frac{\mu_{sec2 \rightarrow sec1}}{\mu_{sec1 \rightarrow tma1}} p_{sec2 \rightarrow sec1 \ sec1 \rightarrow tma1} \end{aligned}$$

$$R_{SEC2,SEC1}(sec2 \rightarrow sec3, S \cap C_{TMA1}) = 0$$

$$R_{SEC2,SEC1}(sec2 \rightarrow tma2, S \cap C_{TMA1}) = 0$$

Le  $m$ -pic d'intensité de trafic de  $S'$  dans  $S \cap C_m$  noté  $\bar{R}(S', S \cap C_m)$ , avec  $S' \subseteq \mathcal{N} - C_m$  est donné par la relation suivante:

$$\bar{R}_m(S', S \cap C_m) = \sum_{m' \in \mu(S')} \max_{i \in S' \cap C_{m'}} R_{m',m}(i, S \cap C_m)$$

$m = GATE$

$$\begin{aligned} \bar{R}_{GATE}(S', S \cap C_{GATE}) &= \sum_{m' \in \mu(S')} \max_{i \in S' \cap C_{m'}} R_{m',GATE}(i, S \cap C_{GATE}) \\ &= \max_{i \in S' \cap C_{TWY}} R_{TWY,GATE}(i, S \cap C_{GATE}) \\ &= R_{TWY,GATE}(twy \rightarrow gate, S \cap C_{GATE}) \end{aligned}$$

$m = TWY$

$$\begin{aligned} \bar{R}_{TWY}(S', S \cap C_{TWY}) &= \sum_{m' \in \mu(S')} \max_{i \in S' \cap C_{m'}} R_{m',TWY}(i, S \cap C_{TWY}) \\ &= \max_{i \in S' \cap C_{GATE}} R_{GATE,TWY}(i, S \cap C_{TWY}) + \max_{i \in S' \cap C_{RWY}} R_{RWY,TWY}(i, S \cap C_{TWY}) \\ &= \max\{R_{GATE,TWY}(emb, S \cap C_{TWY}), R_{GATE,TWY}(deb, S \cap C_{TWY})\} \\ &\quad + R_{RWY,TWY}(att, S \cap C_{TWY}) \end{aligned}$$



$m = RWY$

$$\begin{aligned}\bar{R}_{RWY}(S', S \cap C_{RWY}) &= \sum_{m' \in \mu(S')} \max_{i \in S' \cap C_{m'}} R_{m', RWY}(i, S \cap C_{RWY}) \\ &= \max_{i \in S' \cap C_{TWY}} R_{TWY, RWY}(i, S \cap C_{RWY}) + \max_{i \in S' \cap C_{TMA}} R_{TMA, RWY}(i, S \cap C_{RWY}) \\ &= R_{TWY, RWY}(twy \rightarrow rwy, S \cap C_{RWY}) + R_{TMA, RWY}(tma \rightarrow rwy, S \cap C_{RWY})\end{aligned}$$

$m = TMA$

$$\begin{aligned}\bar{R}_{TMA}(S', S \cap C_{TMA}) &= \sum_{m' \in \mu(S')} \max_{i \in S' \cap C_{m'}} R_{m', TMA}(i, S \cap C_{TMA}) \\ &= \max_{i \in S' \cap C_{SEC}} R_{SEC, TMA}(i, S \cap C_{TMA}) + \max_{i \in S' \cap C_{RWY}} R_{RWY, TMA}(i, S \cap C_{TMA}) \\ &= R_{SEC, TMA}(sec \rightarrow tma, S \cap C_{TMA}) + R_{RWY, TMA}(dec, S \cap C_{TMA})\end{aligned}$$

$m = SEC1$

$$\begin{aligned}\bar{R}_{SEC1}(S', S \cap C_{SEC1}) &= \sum_{m' \in \mu(S')} \max_{i \in S' \cap C_{m'}} R_{m', SEC1}(i, S \cap C_{SEC1}) \\ &= \max_{i \in S' \cap C_{TMA}} R_{TMA, SEC1}(i, S \cap C_{SEC1}) + \max_{i \in S' \cap C_{SEC2}} R_{SEC2, SEC1}(i, S \cap C_{SEC1}) \\ &\quad + \max_{i \in S' \cap C_{SEC3}} R_{SEC3, SEC1}(i, S \cap C_{SEC1}) \\ &= R_{TMA, SEC1}(tma1 \rightarrow sec1, S \cap C_{SEC1}) + R_{SEC2, SEC1}(sec2 \rightarrow sec1, S \cap C_{SEC1}) \\ &\quad + R_{SEC3, SEC1}(sec3 \rightarrow sec1, S \cap C_{SEC1})\end{aligned}$$

**Théorème 5** (Lois de conservation approximatives pour le modèle MQNET). *Sous la condition globale de stabilité (4.12), le modèle MQNET satisfait les lois de conservation approximatives avec les paramètres  $\mathbf{A}$ ,  $\mathbf{b}$  et  $\Phi$  donnés par les relations suivantes:*

$$A_i^S = \begin{cases} V_i^{S \cap C_m}, & \text{si } i \in S \cap C_m \\ 0, & \text{sinon} \end{cases}$$

$$b(S) = \sum_{m \in \mu(S)} \hat{b}(S \cap C_m);$$

et 
$$\Phi(S) = \sum_{m \in \mu(S)} \frac{\bar{R}(\mathcal{N} - C_m, S \cap C_m)}{1 - \rho_0(S \cap C_m) - \bar{R}_m(\mathcal{N} - C_m, S \cap C_m)} \hat{b}(S \cap C_m)$$

où 
$$\hat{b}(S) = \frac{1}{1 - \rho^o(S)} \sum_{j \in S} \rho_j V_j^S$$

Appliquons le Théorème 5.

$$\hat{b}(S \cap C_m) = \frac{1}{1 - \rho^o(S \cap C_m)} \sum_{j \in S \cap C_m} \rho_j V_j^{S \cap C_m}$$

Supposons  $S = \mathcal{A}$

$$\hat{b}(S \cap C_m) = \hat{b}(C_m) = \frac{1}{1 - \rho^o(C_m)} \sum_{j \in C_m} \rho_j V_j^{C_m}$$

$$\begin{aligned} \hat{b}(C_{GATE}) &= \frac{1}{1 - \rho^o(C_{GATE})} \sum_{j \in C_{GATE}} \rho_j V_j^{C_{GATE}} \\ &= \frac{1}{1 - \rho^o(C_{GATE})} (\rho_{emb} V_{emb}^{C_{GATE}} + \rho_{deb} V_{deb}^{C_{GATE}}) \end{aligned}$$

$$\hat{b}(C_{TWY}) = \rho_{twy \rightarrow rwy} V_{twy \rightarrow rwy}^{C_{TWY}} + \rho_{twy \rightarrow gate} V_{twy \rightarrow gate}^{C_{TWY}}$$

$$\hat{b}(C_{RWY}) = \rho_{dec} V_{dec}^{C_{RWY}} + \rho_{att} V_{att}^{C_{RWY}}$$

$$\hat{b}(C_{TMA}) = \rho_{tma \rightarrow rwy} V_{tma \rightarrow rwy}^{C_{TMA}} + \rho_{tma \rightarrow sec} V_{tma \rightarrow sec}^{C_{TMA}}$$

$$\hat{b}(C_{SEC1}) = \frac{1}{1 - \rho^o(C_{SEC1})} (\rho_{sec1 \rightarrow o} V_{sec1 \rightarrow o}^{C_{SEC1}} + \rho_{sec1 \rightarrow sec2} V_{sec1 \rightarrow sec2}^{C_{SEC1}} + \rho_{sec1 \rightarrow sec3} V_{sec1 \rightarrow sec3}^{C_{SEC1}})$$

$$b(S) = \sum_{m \in \mu(S)} \hat{b}(S \cap C_m)$$

$$\begin{aligned} b(S) &= \hat{b}(C_{GATE1}) + \hat{b}(C_{GATE2}) + \hat{b}(C_{GATE3}) \\ &\quad + \hat{b}(C_{TWY1}) + \hat{b}(C_{TWY2}) + \hat{b}(C_{TWY3}) \\ &\quad + \hat{b}(C_{RWY1}) + \hat{b}(C_{RWY2}) + \hat{b}(C_{RWY3}) \\ &\quad + \hat{b}(C_{TMA1}) + \hat{b}(C_{TMA2}) + \hat{b}(C_{TMA3}) \\ &\quad + \hat{b}(C_{SEC1}) + \hat{b}(C_{SEC2}) + \hat{b}(C_{SEC3}) \end{aligned}$$



$$\Phi(S) = \sum_{m \in \mu(S)} \frac{\bar{R}(\mathcal{N} - C_m, S \cap C_m)}{1 - \rho^o(S \cap C_m) - \bar{R}_m(\mathcal{N} - C_m, S \cap C_m)} \hat{b}(S \cap C_m)$$

$$\text{Posons } \hat{d}(S \cap C_m) = \frac{\bar{R}(\mathcal{N} - C_m, S \cap C_m)}{1 - \rho^o(S \cap C_m) - \bar{R}_m(\mathcal{N} - C_m, S \cap C_m)} \hat{b}(S \cap C_m)$$

$$\hat{d}(C_{GATE}) = \frac{\bar{R}_{GATE}(S', C_{GATE})}{1 - \rho^o(C_{GATE}) - \bar{R}_{GATE}(S', C_{GATE})} \hat{b}(C_{GATE})$$

$$\hat{d}(C_{TWY}) = \frac{\bar{R}_{TWY}(S', C_{TWY})}{1 - \bar{R}_{TWY}(S', C_{TWY})} \hat{b}(C_{TWY})$$

$$\hat{d}(C_{RWY}) = \frac{\bar{R}_{RWY}(S', C_{RWY})}{1 - \bar{R}_{RWY}(S', C_{RWY})} \hat{b}(C_{RWY})$$

$$\hat{d}(C_{TMA}) = \frac{\bar{R}_{TMA}(S', C_{TMA})}{1 - \bar{R}_{TMA}(S', C_{TMA})} \hat{b}(C_{TMA})$$

$$\hat{d}(C_{SEC1}) = \frac{\bar{R}_{SEC1}(S', C_{SEC1})}{1 - \rho^o(C_{SEC1}) - \bar{R}_{SEC1}(S', C_{SEC1})} \hat{b}(C_{SEC1})$$

$$\begin{aligned} \Phi(S) = & \hat{d}(C_{GATE1}) + \hat{d}(C_{GATE2}) + \hat{d}(C_{GATE3}) \\ & + \hat{d}(C_{TWY1}) + \hat{d}(C_{TWY2}) + \hat{d}(C_{TWY3}) \\ & + \hat{d}(C_{RWY1}) + \hat{d}(C_{RWY2}) + \hat{d}(C_{RWY3}) \\ & + \hat{d}(C_{TMA1}) + \hat{d}(C_{TMA2}) + \hat{d}(C_{TMA3}) \\ & + \hat{d}(C_{SEC1}) + \hat{d}(C_{SEC2}) + \hat{d}(C_{SEC3}) \end{aligned}$$

À présent que nous avons les relations permettant de calculer  $\Phi(S)$  et  $b(S)$ , nous pouvons appliquer le Théorème 4 (Garantie performance).

En effet, puisque la valeur de  $Z^{OPT}$  est une valeur idéale, il faut la coincer dans un intervalle restreint.

$$Z^D \leq Z^{LP} \leq Z^{OPT} \leq Z^{INDEX}$$

$$Z^{INDEX} \leq Z^{OPT} + \varepsilon \text{ et } Z^D \geq Z^{OPT} - \varepsilon$$

$$\text{où } \varepsilon = \gamma_1 \Phi(\{1, \dots, N\}) + (\gamma_2 - \gamma_1) \Phi(\{2, \dots, N\}) + \dots + (\gamma_N - \gamma_{N-1}) \Phi(\{N\})$$

$Z^{INDEX}$  et  $Z^D$  sont donnés par l'algorithme de Klimov adapté

$$\text{En outre } Z^{INDEX} \leq r Z^{OPT} \text{ et } Z^D \geq \frac{1}{r} Z^{OPT}$$

$$\text{où } r = 1 + \max_{S \subseteq \mathcal{A}} \frac{\Phi(S)}{b(S)}$$

$$\text{On peut établir que } r \leq 1 + \max_{m \in \mathcal{M}} \frac{\bar{R}_m(\mathcal{A} - C_m, C_m)}{1 - \bar{\rho}(m)}$$

## 5.4 Analyse de performance des règles de priorité par les lois de conservation approximatives

Ce paragraphe vise à analyser la performance de certaines règles de priorité  $\pi$ .  $\mathbf{x}^\pi$  est la performance et  $Z^\pi$  est l'objectif.

En analysant la performance, nous cherchons à résoudre deux problèmes: l'évaluation de  $Z^\pi$ , d'une part et le développement des garanties de performance pour  $\pi$  sous la forme de borne pour  $Z^\pi - Z^{OPT}$ , d'autre part.

Soient  $S_k^\pi = \{\pi_k, \dots, \pi_N\}$  les  $(N - k + 1)$  plus grandes classes de priorité sous  $\pi$ . Cet ensemble est donné par l'algorithme de Klimov. Les éléments sont classés par ordre de priorité croissante.  $\pi_{36}$  a la priorité la plus grande.

$$S_1^\pi = \{\pi_1, \dots, \pi_{36}\} = \mathcal{A}$$

$$S_2^\pi = \{\pi_2, \dots, \pi_{36}\} = \mathcal{A} - \{\pi_1\}$$

$$S_3^\pi = \{\pi_3, \dots, \pi_{36}\} = \mathcal{A} - \{\pi_1, \pi_2\}$$

...

$$S_{36}^\pi = \{\pi_{36}\} = \mathcal{A} - \{\pi_1, \dots, \pi_{35}\}$$



$y^\pi$  est la solution du système d'équations linéaires  $\sum_{i=1}^k A_k^{S_i^\pi} y_i^\pi = c_k$ , si  $1 \leq k \leq N$ .

$$A_1^{S_1^\pi} y_1^\pi = c_1$$

$$A_2^{S_1^\pi} y_1^\pi + A_2^{S_2^\pi} y_2^\pi = c_2$$

$$A_3^{S_1^\pi} y_1^\pi + A_3^{S_2^\pi} y_2^\pi + A_3^{S_3^\pi} y_3^\pi = c_3$$

...

$$A_k^{S_1^\pi} y_1^\pi + \dots + A_k^{S_i^\pi} y_i^\pi = c_k$$

Supposons  $dec1$ ,  $dec2$  et  $dec3$  les trois classes les moins prioritaires.  $\pi_1=dec1$ ,  $\pi_2=dec2$ ,  $\pi_3=dec3$

$$A_1^{S_1^\pi} y_1^\pi = c_1 \Leftrightarrow y_1^\pi = \frac{c_1}{A_1^{S_1^\pi}}$$

$$\Rightarrow y_1^\pi = \frac{c_1}{V_1^{S_1^\pi \cap C_m}} = \mu_1 c_1$$

$$A_2^{S_1^\pi} y_1^\pi + A_2^{S_2^\pi} y_2^\pi = c_2 \Rightarrow \frac{1}{\mu_2} \mu_1 c_1 + \frac{1}{\mu_2} y_2^\pi = c_2$$

$$\Leftrightarrow y_2^\pi = \mu_2 c_2 - \mu_1 c_1$$

$$A_3^{S_1^\pi} y_1^\pi + A_3^{S_2^\pi} y_2^\pi + A_3^{S_3^\pi} y_3^\pi = c_3 \Rightarrow \frac{1}{\mu_3} \mu_1 c_1 + \frac{1}{\mu_3} (\mu_2 c_2 - \mu_1 c_1) + \frac{1}{\mu_3} y_3^\pi = c_3$$

$$\Leftrightarrow y_3^\pi = \mu_3 c_3 - \mu_2 c_2$$

**Théorème 7** (Évaluation de  $Z^\pi$ ) Sous la condition de stabilité globale (4.12)

$$-\sum_{k=1}^N (y_k^\pi)^- \Phi(S_k^\pi) \leq Z^\pi - \sum_{k=1}^N y_k^\pi b(S_k^\pi) \leq \sum_{k=1}^N (y_k^\pi) \Phi(S_k^\pi) \quad (4.51)$$

Le Théorème 7 sera appliqué lors de l'implémentation dans la mesure où l'algorithme de Klimov nous permet d'obtenir un ensemble  $S_k^\pi$  concret. En outre, le calcul de  $\Phi(S)$  et de  $b(S)$  implique de nombreuses opérations.

Considérons  $\tilde{\mathbf{P}}$  une modification de la matrice de feed-back  $\mathbf{P}$  en faisant une estimation des flux entre des stations distinctes grâce aux départs des avions du système.

$$\tilde{\mathbf{P}} = \begin{cases} p_{ij} & \text{si } s(i) = s(j) \\ 0 & \text{sinon} \end{cases} \quad \text{avec } p_{io} = 1 - \sum_{j \in \mathcal{N}} \tilde{p}_{ij}$$

Nous utilisons par la suite le symbole  $\sim$  pour noter les quantités relatives au système modifié.

Il est important de signaler que  $\alpha$ ,  $\mu$  et la matrice  $\mathbf{A}$  ne sont pas modifiés.

$$\tilde{b}(S) = \sum_{m \in \mu(S)} \left\{ \frac{\sum_{j \in S \cap C_m} \tilde{\rho}_j V_j^{S \cap C_m}}{1 - \sum_{j \in S \cap C_m} \alpha_j V_j^{S \cap C_m}} \right\}$$

$$\text{où } \tilde{\rho}_j = \frac{\tilde{\lambda}_j}{\mu_j}, j \in \mathcal{N}.$$

$\tilde{\lambda}$  est modifié, car il s'exprime en fonction des probabilités de transition.

Considérons le cas général  $S = \mathcal{N}$  et calculons  $\tilde{b}(S)$  sachant que dans ce cas  $\mu(S) = \mathcal{M}$ .

$$\text{Posons } \hat{f}(S \cap C_m) = \frac{\sum_{j \in S \cap C_m} \tilde{\rho}_j V_j^{S \cap C_m}}{1 - \sum_{j \in S \cap C_m} \alpha_j V_j^{S \cap C_m}}$$

$$\begin{aligned} \hat{f}(S \cap C_{GATE}) &= \frac{\sum_{j \in S \cap C_{GATE}} \tilde{\rho}_j V_j^{S \cap C_{GATE}}}{1 - \sum_{j \in S \cap C_{GATE}} \alpha_j V_j^{S \cap C_{GATE}}} \\ &= \frac{\tilde{\rho}_{emb} V_{emb}^{C_{GATE}} + \tilde{\rho}_{deb} V_{deb}^{C_{GATE}}}{1 - \alpha_{emb} V_{emb}^{C_{GATE}}} \end{aligned}$$

$$\hat{f}(S \cap C_{TWY}) = \tilde{\rho}_{twy \rightarrow gate} V_{twy \rightarrow gate}^{C_{TWY}} + \tilde{\rho}_{twy \rightarrow rwy} V_{twy \rightarrow rwy}^{C_{TWY}}$$

$$\hat{f}(S \cap C_{RWY}) = \tilde{\rho}_{all} V_{all}^{C_{RWY}} + \tilde{\rho}_{dec} V_{dec}^{C_{RWY}}$$



$$\hat{f}(S \cap C_{TMA}) = \tilde{\rho}_{tma \rightarrow sec} V_{tma \rightarrow sec}^{C_{TMA}} + \tilde{\rho}_{tma \rightarrow rwy} V_{tma \rightarrow rwy}^{C_{TMA}}$$

$$\hat{f}(S \cap C_{SEC1}) = \frac{\tilde{\rho}_{sec1 \rightarrow o} V_{sec1 \rightarrow o}^{C_{SEC1}} + \tilde{\rho}_{sec1 \rightarrow sec2} V_{sec1 \rightarrow sec2}^{C_{SEC1}} + \tilde{\rho}_{sec1 \rightarrow sec3} V_{sec1 \rightarrow sec3}^{C_{SEC1}} + \tilde{\rho}_{sec1 \rightarrow tma1} V_{sec1 \rightarrow tma1}^{C_{SEC1}}}{1 - \alpha_{sec1 \rightarrow sec2} V_{sec1 \rightarrow sec2}^{C_{SEC1}} - \alpha_{sec1 \rightarrow sec3} V_{sec1 \rightarrow sec3}^{C_{SEC1}} - \alpha_{sec1 \rightarrow tma1} V_{sec1 \rightarrow tma1}^{C_{SEC1}}}$$

$$\begin{aligned} \tilde{b}(S) &= \sum_{m \in \mu(S)} \left\{ \frac{\sum_{j \in S \cap C_m} \tilde{\rho}_j V_j^{S \cap C_m}}{1 - \sum_{j \in S \cap C_m} \alpha_j V_j^{S \cap C_m}} \right\} \\ &= \hat{f}(S \cap C_{SEC1}) + \hat{f}(S \cap C_{SEC2}) + \hat{f}(S \cap C_{SEC3}) \\ &\quad + \hat{f}(S \cap C_{TMA1}) + \hat{f}(S \cap C_{TMA2}) + \hat{f}(S \cap C_{TMA3}) \\ &\quad + \hat{f}(S \cap C_{RWY1}) + \hat{f}(S \cap C_{RWY2}) + \hat{f}(S \cap C_{RWY3}) \\ &\quad + \hat{f}(S \cap C_{TWY1}) + \hat{f}(S \cap C_{TWY2}) + \hat{f}(S \cap C_{TWY3}) \\ &\quad + \hat{f}(S \cap C_{GATE1}) + \hat{f}(S \cap C_{GATE2}) + \hat{f}(S \cap C_{GATE3}) \end{aligned}$$

$$\Delta(S) = b(S) - \tilde{b}(S) = \sum_{m \in \mu(S)} \left\{ \frac{\sum_{j \in S \cap C_m} (\rho_j - \tilde{\rho}_j) V_j^{S \cap C_m}}{1 - \rho^0(S \cap C_m)} \right\}$$

Calculons la quantité  $\Delta(S)$

$$\text{Posons } \hat{g}(S \cap C_m) = \frac{\sum_{j \in S \cap C_m} (\rho_j - \tilde{\rho}_j) V_j^{S \cap C_m}}{1 - \rho^0(S \cap C_m)}$$

$$\begin{aligned} \hat{g}(S \cap C_{GATE}) &= \frac{\sum_{j \in S \cap C_{GATE}} (\rho_j - \tilde{\rho}_j) V_j^{S \cap C_{GATE}}}{1 - \rho^0(S \cap C_{GATE})} \\ &= \frac{(\rho_{emb} - \tilde{\rho}_{emb}) V_{emb}^{S \cap C_{GATE}} + (\rho_{deb} - \tilde{\rho}_{deb}) V_{deb}^{S \cap C_{GATE}}}{1 - \alpha_{emb} V_{emb}^{S \cap C_{GATE}}} \end{aligned}$$

$$\hat{g}(S \cap C_{TWY}) = (\rho_{twy \rightarrow rwy} - \tilde{\rho}_{twy \rightarrow rwy}) V_{twy \rightarrow rwy}^{S \cap C_{TWY}} + (\rho_{twy \rightarrow gate} - \tilde{\rho}_{twy \rightarrow gate}) V_{twy \rightarrow gate}^{S \cap C_{TWY}}$$

$$\hat{g}(S \cap C_{RWY}) = (\rho_{att} - \tilde{\rho}_{att}) V_{att}^{S \cap C_{RWY}} + (\rho_{dec} - \tilde{\rho}_{dec}) V_{dec}^{S \cap C_{RWY}}$$

$$\hat{g}(S \cap C_{TMA}) = (\rho_{tma \rightarrow rwy} - \tilde{\rho}_{tma \rightarrow rwy}) V_{tma \rightarrow rwy}^{S \cap C_{TMA}} + (\rho_{tma \rightarrow sec} - \tilde{\rho}_{tma \rightarrow sec}) V_{tma \rightarrow sec}^{S \cap C_{TMA}}$$

$$\begin{aligned} \hat{g}(S \cap C_{SEC1}) = & \frac{(\rho_{sec1 \rightarrow o} - \tilde{\rho}_{sec1 \rightarrow o})V_{sec1 \rightarrow o}^{S \cap C_{SEC1}} + (\rho_{sec1 \rightarrow sec2} - \tilde{\rho}_{sec1 \rightarrow sec2})V_{sec1 \rightarrow sec2}^{S \cap C_{SEC1}}}{1 - \alpha_{sec1 \rightarrow sec2}V_{sec1 \rightarrow sec2}^{S \cap C_{SEC1}} - \alpha_{sec1 \rightarrow sec3}V_{sec1 \rightarrow sec3}^{S \cap C_{SEC1}} - \alpha_{sec1 \rightarrow tmal}V_{sec1 \rightarrow tmal}^{S \cap C_{SEC1}}} \\ & + \frac{(\rho_{sec1 \rightarrow sec3} - \tilde{\rho}_{sec1 \rightarrow sec3})V_{sec1 \rightarrow sec3}^{S \cap C_{SEC1}} + (\rho_{sec1 \rightarrow tmal} - \tilde{\rho}_{sec1 \rightarrow tmal})V_{sec1 \rightarrow tmal}^{S \cap C_{SEC1}}}{1 - \alpha_{sec1 \rightarrow sec2}V_{sec1 \rightarrow sec2}^{S \cap C_{SEC1}} - \alpha_{sec1 \rightarrow sec3}V_{sec1 \rightarrow sec3}^{S \cap C_{SEC1}} - \alpha_{sec1 \rightarrow tmal}V_{sec1 \rightarrow tmal}^{S \cap C_{SEC1}}} \end{aligned}$$

$$\begin{aligned} \Delta(S) = & \hat{g}(S \cap C_{GATE1}) + \hat{g}(S \cap C_{GATE2}) + \hat{g}(S \cap C_{GATE3}) \\ & + \hat{g}(S \cap C_{TWY1}) + \hat{g}(S \cap C_{TWY2}) + \hat{g}(S \cap C_{TWY3}) \\ & + \hat{g}(S \cap C_{RWY1}) + \hat{g}(S \cap C_{RWY2}) + \hat{g}(S \cap C_{RWY3}) \\ & + \hat{g}(S \cap C_{TMA1}) + \hat{g}(S \cap C_{TMA2}) + \hat{g}(S \cap C_{TMA3}) \\ & + \hat{g}(S \cap C_{SEC1}) + \hat{g}(S \cap C_{SEC2}) + \hat{g}(S \cap C_{SEC3}) \end{aligned}$$

Il résulte du Théorème 5 que  $\Phi \equiv 0$ . De ce fait, on déduit du Théorème 7 que

$$\sum_{k=1}^N y_k^\pi \tilde{b}(S_k^\pi) = \tilde{Z}^\pi,$$

qui représente le coût encouru quand on contrôle le réseau modifié par  $\pi$ .

Afin d'obtenir des résultats concrets de  $\tilde{Z}^\pi$  pour le réseau étudié, il faut les résultats fournis par l'algorithme de Klimov. Nous calculerons ainsi  $\tilde{Z}^\pi$  et l'application des corollaires 2 et 3 lors de l'implémentation.

## Corollaire 2

$$\Delta_1(\pi; \mathbf{P}, \tilde{\mathbf{P}}) \leq Z^\pi - \tilde{Z}^\pi \leq \Delta_2(\pi; \mathbf{P}, \tilde{\mathbf{P}}),$$

où

$$\Delta_1(\pi; \mathbf{P}, \tilde{\mathbf{P}}) = -\sum_{k=1}^N (y_k^\pi)^- \Phi(S_k^\pi) + \sum_{k=1}^N y_k^\pi \Delta(S_k^\pi),$$

et

$$\Delta_2(\pi; \mathbf{P}, \tilde{\mathbf{P}}) = -\sum_{k=1}^N (y_k^\pi)^+ \Phi(S_k^\pi) + \sum_{k=1}^N y_k^\pi \Delta(S_k^\pi).$$



**Corollaire 2** (Borne de suboptimisation pour  $\pi$ )

$$Z^\pi - Z^{OPT} \leq (Z^{INDEX} - Z^{OPT}) + (\tilde{Z}^\pi - \tilde{Z}^{INDEX}) \\ + \Delta_2(\pi; \mathbf{P}, \tilde{\mathbf{P}}) - \Delta_2(INDEX; \mathbf{P}, \tilde{\mathbf{P}}).$$

## Chapitre 6

### Implémentation du modèle MQNET

Pour implémenter le modèle MQNET proposé pour la navigation aérienne, nous allons utiliser le *Pascal* comme langage de programmation.

#### Définition des types

Serveur = (GATE1, GATE2, GATE3, TWY1, TWY2, TWY3, RWY1, RWY2, RWY3, TMA1, TMA2, TMA3, SEC1, SEC2, SEC3)

Type énuméré; énumération des différents serveurs du réseau.

Client = (emb1, emb2, emb3, deb1, deb2, deb3, twy1\_gate1, twy1\_rwy1, twy2\_gate2, twy2\_rwy2, twy3\_gate3, twy3\_rwy3, att1, dec1, att2, dec2, att3, dec3, tma1\_rwy1, tma1\_sec1, tma2\_rwy2, tma2\_sec2, tma3\_rwy3, tma3\_sec3, sec1\_o, sec1\_sec2, sec1\_sec3, sec1\_tma1, sec2\_o, sec2\_sec1, sec2\_sec3, sec2\_tma2, sec3\_o, sec3\_sec1, sec3\_sec2, sec3\_tma3)

Type énuméré; énumération des différents clients du réseau

Station = set of Serveur

*Sous ensemble de l'ensemble des serveurs*

Avion = set of Client

*Sous ensembles de l'ensemble des clients*

TabClasse=array[emb1..sec3\_tma3]of integer

*Tableau indexé par les classes d'avions, les éléments sont des entiers*

TabClasseR=array[emb1..sec3\_tma3]of real

*Tableau indexé par les classes d'avions, les éléments sont des réels*

TabServeur=array[GATE1..SEC3] of real

*Tableau indexé par les serveurs, les éléments sont des réels*

MatriceProba=array[emb1..sec3\_tma3, emb1..sec3\_tma3]of real

*Tableau à deux dimensions qui contiendra les probabilité de transition*

TypePi = array[1..36]of Client

TypeGammaPi = array[1..36]of real



## Définition des constantes

M = 15

*Nombre de serveurs*

N = 36

*Nombre de classes d'avions*

StationsDeService: Station = [GATE1, GATE2, GATE3, TWY1, TWY2, TWY3, RWY1, RWY2, RWY3, TMA1, TMA2, TMA3, SEC1, SEC2, SEC3]

*Ensemble des différentes stations de service.*

ClassesDAvions: Avion=[emb1, emb2, emb3, deb1, deb2, deb3, twy1\_gate1, twy1\_rwy1, twy2\_gate2, twy2\_rwy2, twy3\_gate3, twy3\_rwy3, att1, dec1, att2, dec2, att3, dec3, tma1\_rwy1, tma1\_sec1, tma2\_rwy2, tma2\_sec2, tma3\_rwy3, tma3\_sec3, sec1\_o, sec1\_sec2, sec1\_sec3, sec1\_tma1, sec2\_o, sec2\_sec1, sec2\_sec3, sec2\_tma2, sec3\_o, sec3\_sec1, sec3\_sec2, sec3\_tma3]

*Ensemble des différentes classes d'avions.*

C\_GATE1:Avion=[emb1,deb1]

C\_GATE2:Avion=[emb2,deb2]

C\_GATE3:Avion=[emb3,deb3]

C\_TWY1:Avion=[twy1\_rwy1,twy1\_gate1]

C\_TWY2:Avion=[twy2\_rwy2,twy2\_gate2]

C\_TWY3:Avion=[twy3\_rwy3,twy3\_gate3]

C\_RWY1:Avion=[att1,dec1]

C\_RWY2:Avion=[att2,dec2]

C\_RWY3:Avion=[att3,dec3]

C\_TMA1:Avion=[tma1\_rwy1,tma1\_sec1]

C\_TMA2:Avion=[tma2\_rwy2,tma2\_sec2]

C\_TMA3:Avion=[tma3\_rwy3,tma3\_sec3]

C\_SEC1:Avion=[Sec1\_o,sec1\_sec2,sec1\_sec3,sec1\_tma1]

C\_SEC2:Avion=[Sec2\_o,sec2\_sec1,sec2\_sec3,sec2\_tma2]

C\_SEC3:Avion=[Sec3\_o,sec3\_sec1,sec3\_sec2,sec3\_tma3]

*Chaque station de service dessert un ensemble constant de classes d'avions*

AvionExterieur: Avion = [emb1, emb2, emb3, sec1\_sec2, sec1\_sec3, sec1\_tma1, sec2\_sec1, sec2\_sec3, sec2\_tma2, sec3\_sec1, sec3\_sec2, sec3\_tma3]

*Ensemble des classes d'avions susceptibles de venir de l'extérieur du réseau*

## Définition des variables

Alpha: TabClasse

*Tableau des taux d'arrivée de l'extérieur des différentes classes d'avions*

Mu: TabClasse

*Tableau des taux du temps de service des différentes classes d'avions*

Lambda: TabClasseR

*Tableau des taux d'arrivée totaux des différentes classes d'avions*

Proba: MatriceProba

*Matrice des probabilités de transition d'une classe à une autres*

V\_Cm: TabClasseR

*Tableau des S-Charge de travail moyenne de chaque classe d'avions par rapport à toutes les classes desservies dans la station de service.*

V\_Classe: TabClasseR

*Tableau des S-Charge de travail moyenne de chaque classe d'avions par rapport à toutes les classes desservies dans la station de service.*

Rho\_j: TabClasseR

*Tableau des intensités du trafic nominal*

RhoCm: TabServeur

*Tableau des intensités du trafic nominal au niveau de chaque station de service*

RhoBarreM: TabServeur

*Tableau des intensités du trafic pour chaque station de service*

Cout: TabClasse

*Tableau des coûts encourus par les différentes classes d'avions*

Pi: TypePi

*Tableau des différentes permutations des classes d'avions, résultat de l'algorithme de Klimov*

GammaPi: TypeGammaPi

*Vecteur d'indices de priorité, résultat de l'algorithme de Klimov*

Y\_BarreS: TypeGammaPi

*Résultat de l'algorithme de Klimov, ce tableau enregistrera les  $\bar{y}(S)$*

S: array[1..N]of Avion;

*Tableau dont les élément sont des sous-ensembles de l'ensemble des classes d'avions*

### Définition des procédures et des fonctions

procedure DONNER\_LES\_ALPHA(var T:TabClasse);

*Cette procédure permet à l'utilisateur de donner au programme les différentes valeurs du taux d'arrivée dans le réseau des différentes classes d'avions.*

Argument: -

Résultat: tableau T garni de valeurs

procedure GARNIR\_LES\_MU(var T:TabClasse);

*Cette procédure permet à l'utilisateur de donner au programme les différentes valeurs du taux de service des différentes classes d'avions.*

Argument: -

Résultat: tableau T garni de valeurs

procedure GARNIR\_PROBA\_TRANSITION(var T:MatriceProba);

) 2



*Cette procédure permet à l'utilisateur de donner au programme les différentes valeurs des probabilités de transition d'une classe vers une autre classe.*

Argument: -

Résultat: matrice T garnie de valeur

procedure CALCULER\_LES\_LAMBDA(var T:TabClasseR)

*Cette procédure calcule les différentes valeurs des taux d'arrivée totaux. Ces valeurs sont enregistrées dans le tableau T. Elle utilise les variables globales Alpha et Proba.*

Argument: -

Résultat: tableau T garni de valeurs

procedure DETERMINER\_Cm(m:Serveur;var EnsembleDeCleints:Avion)

*Cette procédure permet de déterminer l'ensemble des classes d'avions desservies à une station de service donnée.*

Argument: un serveur m

Résultat: l'ensemble EnsembleDeClients

procedure CALCULER\_LES\_RHOj(T1:TabClasseR;T2:TabClasse;var T3:TabClasseR);

*Cette procédure permet de calculer l'intensité du trafic nominal des j-avions.*

Argument: tableau T1 contenant les taux d'arrivée totaux, tableau T2 contenant les taux de service.

Résultat: tableau-T3 contenant les valeurs de l'intensité du trafic nominal

procedure CALCULER\_LES\_Vj\_cm(T1:TabClasse;M:MatriceProba;var T2:TabClasseR)

*Cette procédure permet de calculer la S-charge de travail moyenne de chaque classe d'avions*

Argument: T1 des taux de service, M matrice des probabilité

Résultat: T2 des S-charge de travail moyenne par rapport à tous les clients de la station.

procedure CALCULER\_LES\_V\_Classe(T1:TabClasse;var T2:TabClasseR)

*cette procédure permet de calculer la S-charge de travail moyenne par rapport au client lui-même.*

Argument: T1 des taux de service

Résultat: T2 des S-charge de travail

function RHO\_O\_Cm(m:Serveur):real;

*Cette fonction renvoie la valeur de l'intensité du trafic externe au niveau d'une station de service m donnée*

function RHO\_S(S: Avion): real

*Cette fonction renvoie la valeur de l'intensité du trafic pour un ensemble S d'avions donné*

function Ri\_m(i:Client;m:Serveur):real

*Cette fonction renvoie la valeur du pic d'intensité du trafic de la classe i dans une station de service m.*

function R\_BARRE\_M1\_M2(m1, m2: Serveur):real

*Cette fonction renvoie la valeur du pic d'intensité du trafic d'une station m1 vers une station m2.*

function RHO\_BARRE(m: Serveur):real



*Cette fonction renvoie la valeur du pic d'intensité du trafic pour une station de service m donnée.*

procedure GARNIR\_RHO\_Cm(var T: TabServeur)

*Cette procédure permet de garnir le tableau T des valeurs de l'intensité du trafic nominal au niveau de chaque station de service.*

Argument: -

Résultat: tableau T des intensités du trafic nominal.

procedure GARNIR\_RHO\_BARRE\_M(var T: TabServeur)

*Cette procédure permet de garnir le tableau T des valeurs de pic d'intensité du trafic pour une station de service donnée.*

Argument: -

Résultat: tableau T des pic d'intensité du trafic.

function STABILITE\_GLOBALE: boolean

*Cette fonction renvoie "true" si le réseau est globalement stable et elle renvoie "false" sinon.*

function BORNE\_SUPERIEURE: real

*Cette fonction renvoie la valeur du nombre maximum d'avions qu'on peut avoir dans le réseau à un moment donné. Elle fait appel à une sous fonction MIN\_V\_Cm.*

function MIN\_V\_Cm(S: Avion): real

*Cette fonction renvoie le minimum de la S-charge du trafic d'un ensemble S d'avions donné.*

procedure DONNER\_LES\_COUTS(var T: TabClasse)

*Cette procédure permet à l'utilisateur de donner au programme les différents coûts encourus par chaque classe d'avions.*

Argument: -

Résultat: tableau T contenant les différents coûts.

procedure KLIMOV(var Pi: TypePi; var GammaPi: TypeGammaPi; var Y\_BarreS: TypeGammaPi);

*Cette procédure est une implémentation de l'algorithme adapté de Klimov. Elle utilise les variables globales Cout et V\_Cm. Elle garnie les variables Pi, GammaPi et Y\_BarreS. Elle fait appel à deux sous procédures RECHERCHER\_MINIMUM et MAJ\_TAB.*

Argument: -

Résultat: Pi, GammaPi et Y\_Barre\_S

procedure RECHERCHER\_MINIMUM(T: TabClasseR; var Minimum: real; var Position: Client)

*Cette procédure permet de rechercher le minimum des éléments d'un tableau et l'index de ce minimum.*

Argument: tableau T

Résultat: valeur du Minimum et Position de ce minimum.

procedure MAJ\_TAB(var T: TabClasseR)

*Cette procédure permet de mettre à jour le contenu du CsurA déclaré global.*

function R\_M1\_M2(Server1: Serveur; Server2: Serveur; Classe: Client; Ensemble: Avion): real



*Cette fonction renvoie la valeur du  $(m', m)$ -pic d'intensité du trafic de la classe  $i \in C_m$  vers  $S \cap C_m$ .*

function R\_BARRE\_M(Server: Serveur; Ensemble: Avion):real

*Cette fonction renvoie la valeur du  $m$ -pic d'intensité du trafic d'un ensemble  $S'$  vers  $S \cap C_m$*

function B\_CHAPEAU\_DE\_S\_INTER\_Cm(Ensemble:avion; Server:Serveur):real

*Cette fonction renvoie la valeur de  $\hat{b}(S \cap C_m)$*

function PHI\_DE\_S(Ensemble:Avion; m:Serveur):real

*Cette fonction renvoie la valeur de  $\Phi(S)$*

Nous avons repris en annexe de ce travail toutes les procédures et les fonctions qui ont été implémentées à ce jour et dont les tests sont concluants. Mais certaines fonctions telles que STABILITE\_GLOBALE nécessitent un jeu de test cohérentes pour un test exhaustif des autres procédures et fonction en aval. Si le jeu de test n'est pas réaliste, la condition de stabilité globale n'est pas satisfaite et le programme se met à errer et les résultats du programme sont purement et simplement incohérent. La recherche des bonnes valeurs pour le test du modèle est donc cruciale. Ce que nous tâcherons de faire lors de la défense de ce travail dans la mesure où nous ne disposons pas de données réels de la gestion de l'espace aérien.

## Bibliographie

ALLIOT, Jean-Marc. 1998. *Techniques d'optimisation stochastique appliquées à certains problèmes du trafic aérien*. École Nationale de l'Aviation Civile.

BERTSIMAS, D. 1995. *The achievable region in the optimal control of queueing systems; formulations, bounds and policies*. In *Queueing Systems* 21, p.337-389.

BERTSIMAS, D. & J. NINO MORA. 1996. *Optimization of multiclass queueing networks with changeover times via the achievable region approach: Part I, the multi-station case*. Working paper, Operations Research Center, M.I.T.

BERTSIMAS, D. & J. NINO MORA. 1996a. *Conservation laws, extended polymatroids and multi-armed bandit problems; polyhedral approach to indexable systems*. In *Mathematics of Operations Research* 21, p.257-306.

BERTSIMAS, D., I. PASCHALIDIS & J. TSITSIKLIS. 1994. *Optimisation of multiclass queueing networks: polyhedral and nonlinear characterizations of achievable performance*. In *Annals of Applied Probability* 4, p. 43-75.

BURKE, P.J. 1956. *The output of a queueing system*. In *Operations Research* 4, p. 699-704.

BUZACOTT, J.A. & J.G. SHANTHIKUMAR. 1993. *Stochastic Models of Manufacturing Systems*. Prentice Hall, Englewood Cliffs, NJ.

DAI, J.G. & G. WEISS. 1996. *Stability and instability of fluid models for reentrant lines*. In *Mathematics of Operations Research* 21, p. 115-134.

FINCH, P.D. 1959. *On the distribution of queue size in queueing problems*. In *Acta Mathematica Hungarica* 10, p.327-336.

GELENBE, E. & I. MITRANI. 1980. *Analysis and Synthesis of Computer Systems*. Academic Press, London.

GLAZEBROOK K.D. & R. GARBE. 1996. *Reflections on a new approach to Gittins indexation*. In *Journal of the Operational Research Society* 47, p. 1301-1309.

GLAZEBROOK K.D. & R. GARBE. 1997. *Almost optimal policies for systems which almost satisfy conservation laws*. Technical Report, Department of Statistics, Newcastle University.

GOEMANS, M.X. & D.P. WILLIAMSON. 1996. *The primal-dual method for approximation algorithms and its application to network design problems*. In *Approximation Algorithms*, ed. D. Hochbaum, p. 144-191.

HARRISON, J.M. & L.M. WEIN. 1989. *Scheduling networks of queues: heavy traffic analysis of a simple open network*. In *Queueing Systems* 5, p.265-280.

KLIMOV, G.P. 1974. *Time sharing service systems I*. In *Theory of Probability and its Applications* 19, p. 532-551.



- KOBAYASHI, H. 1978. *Modeling and Analysis. An Introduction to System Performance Evaluation Methodology*. Addison-Wesley Publishing Company, Reading, Massachusetts.
- KUMAR, P.R. & S.P. MEYN. 1996. *Duality and linear programs for stability and performance analysis of queueing networks and scheduling policies*. In *IEEE Transactions on Automatic Control* 41, p. 4-16.
- KUMAR, S. & P.R. KUMAR. 1994. *Performance bounds for queueing networks and scheduling policies*. In *IEEE Transactions on Automatic Control* 39, p. 1600-1611.
- MACKAY, Wendy E., Anne-Laure FAYARD & Lionel MÉDINI. *Is Paper Safer? Augmenting Paper Flight Strips to Support Air Traffic Control*.
- PAPADIMITRIOU, C.H. & J.N. TSITSIKLIS. 1993. *The complexity of optimal queueing network control*. Working Paper, Laboratory for Intelligent Decision Systems.
- STUHLIK, Jean-Baptiste. 1999. *Le chaos programmé du ciel européen*. In *La recherche* 319, p. 52-61.
- THOMAS, Paul, Bernard MIAILLER *et alii*. 1997. *ATM R&D Strategy*. Technical report, Eurocontrol.

```

program AVIATION;

uses wincrt;

type Serveur=(GATE1,GATE2,GATE3,
               TWY1,TWY2,TWY3,
               RWY1,RWY2,RWY3,
               TMA1,TMA2,TMA3,
               SEC1,SEC2,SEC3);

Client=(emb1,emb2,emb3,deb1,deb2,deb3,
        twy1_gate1,twy1_rwy1,twy2_gate2,twy2_rwy2,twy3_gate3,twy3_r
wy3,
        att1,dec1,att2,dec2,att3,dec3,
        tma1_rwy1,tma1_sec1,tma2_rwy2,tma2_sec2,tma3_rwy3,tma3_sec3

        sec1_o,sec1_sec2,sec1_sec3,sec1_tma1,
        sec2_o,sec2_sec1,sec2_sec3,sec2_tma2,
        sec3_o,sec3_sec1,sec3_sec2,sec3_tma3);

Station=set of Serveur;

Avion=set of Client;

TabClasse=array[emb1..sec3_tma3]of integer;

TabClasseR=array[emb1..sec3_tma3]of real;

TabServeur=array[GATE1..SEC3] of real;

MatriceProba=array[emb1..sec3_tma3,emb1..sec3_tma3]of real;

TypePi=array[1..36]of Client;

TypeGammaPi=array[1..36]of real;

const M=15;

N=36;

StationsDeService: Station =[GATE1,GATE2,GATE3,
                              TWY1,TWY2,TWY3,
                              RWY1,RWY2,RWY3,
                              TMA1,TMA2,TMA3,
                              SEC1,SEC2,SEC3];

ClassesDAvions: Avion=[emb1,emb2,emb3,deb1,deb2,deb3,
                        twy1_gate1,twy1_rwy1,twy2_gate2,twy2_rwy2,twy3_
gate3,twy3_rwy3,
                        att1,dec1,att2,dec2,att3,dec3,
                        tma1_rwy1,tma1_sec1,tma2_rwy2,tma2_sec2,tma3_rw
y3,tma3_sec3,
                        sec1_o,sec1_sec2,sec1_sec3,sec1_tma1,
                        sec2_o,sec2_sec1,sec2_sec3,sec2_tma2,
                        sec3_o,sec3_sec1,sec3_sec2,sec3_tma3];

C_GATE1:Avion=[emb1,deb1];
C_GATE2:Avion=[emb2,deb2];
C_GATE3:Avion=[emb3,deb3];
C_TWY1:Avion=[twy1_rwy1,twy1_gate1];

```



```

C_TWY2:Avion=[twy2_rwy2,twy2_gate2];
C_TWY3:Avion=[twy3_rwy3,twy3_gate3];
C_RWY1:Avion=[att1,dec1];
C_RWY2:Avion=[att2,dec2];
C_RWY3:Avion=[att3,dec3];
C_TMA1:Avion=[tma1_rwy1,tma1_sec1];
C_TMA2:Avion=[tma2_rwy2,tma2_sec2];
C_TMA3:Avion=[tma3_rwy3,tma3_sec3];
C_SEC1:Avion=[Sec1_o,sec1_sec2,sec1_sec2,sec1_tma1];
C_SEC2:Avion=[Sec2_o,sec2_sec1,sec2_sec3,sec2_tma2];
C_SEC3:Avion=[Sec3_o,sec3_sec1,sec3_sec2,sec3_tma3];

AvionExterieur:Avion=[emb1,emb2,emb3,
                      sec1_sec2,sec1_sec3,sec1_tma1,
                      sec2_sec1,sec2_sec3,sec2_tma2,
                      sec3_sec1,sec3_sec2,sec3_tma3];

var Alpha:TabClasse;
Mu:TabClasse;
Lambda:TabClasseR;
Proba:MatriceProba;
V_Cm:TabClasseR;
V_Classe:TabClasseR;
Rho_j:TabClasseR;
RhoCm:TabServeur;
RhoBarreM:TabServeur;
Cout:TabClasse;
Pi:TypePi;
GammaPi:TypeGammaPi;
Y_BarreS:TypeGammaPi;
S:array[1..N]of Avion;

procedure DONNER_LES_COUTS(var T:TabClasse);
var Cli:Client;
c:integer;
begin
  writeln('Donner les différents coûts encourus');
  writeln('DONNER DES VALEURS ENTIERES, POSITIVES OU NULLES,<=100
S.V.P');
  c:=1;
  for Cli:=emb1 to sec3_tma3 do
  begin
    case c of
      1:write('Cout emb1 = ');
      2:write('Cout emb2 = ');
      3:write('Cout emb3 = ');
      4:write('Cout deb1 = ');
      5:write('Cout deb2 = ');
      6:write('Cout deb3 = ');
      7:write('Cout twy1_gate1 = ');
      8:write('Cout twy1_rwy1 = ');
      9:write('Cout twy2_gate2 = ');
      10:write('Cout twy2_rwy2 = ');
      11:write('Cout twy3_gate3 = ');
      12:write('Cout twy3_rwy3 = ');
      13:write('Cout att1 = ');
      14:write('Cout dec1 = ');
      15:write('Cout att2 = ');
      16:write('Cout dec2 = ');
      17:write('Cout att3 = ');
      18:write('Cout dec3 = ');
    end
  end
end

```

```

19:write('Cout tma1_rwy1 = ');
20:write('Cout tma1_sec1 = ');
21:write('Cout tma2_rwy2 = ');
22:write('Cout tma2_sec2 = ');
23:write('Cout tma3_rwy3 = ');
24:write('Cout tma3_sec3 = ');
25:write('Cout sec1_o = ');
26:write('Cout sec1_sec2 = ');
27:write('Cout sec1_sec3 = ');
28:write('Cout sec1_tma1 = ');
29:write('Cout sec2_o = ');
30:write('Cout sec2_sec1 = ');
31:write('Cout sec2_sec3 = ');
32:write('Cout sec2_tma2 = ');
33:write('Cout sec3_o = ');
34:write('Cout sec3_sec1 = ');
35:write('Cout sec3_sec2 = ');
36:write('Cout sec3_tma3 = ');
end;(*fin case *)
readln(T[Cli]);
c:=succ(c);
end;(*fin boucle for*)
end;(*fin procedure*)

procedure DETERMINER_Cm(m:Serveur;var EnsembleDeCleints:Avion);
begin
  case m of
    GATE1:EnsembleDeCleints:=[emb1,deb1];
    GATE2:EnsembleDeCleints:=[emb2,deb2];
    GATE3:EnsembleDeCleints:=[emb3,deb3];
    TWY1:EnsembleDeCleints:=[twy1_rwy1,twy1_gate1];
    TWY2:EnsembleDeCleints:=[twy2_rwy2,twy2_gate2];
    TWY3:EnsembleDeCleints:=[twy3_rwy3,twy3_gate3];
    RWY1:EnsembleDeCleints:=[att1,dec1];
    RWY2:EnsembleDeCleints:=[att2,dec2];
    RWY3:EnsembleDeCleints:=[att3,dec3];
    TMA1:EnsembleDeCleints:=[tma1_rwy1,tma1_sec1];
    TMA2:EnsembleDeCleints:=[tma2_rwy2,tma2_sec2];
    TMA3:EnsembleDeCleints:=[tma3_rwy3,tma3_sec3];
    SEC1:EnsembleDeCleints:=[Sec1_o,sec1_sec2,sec1_sec3,sec1_tma1];
    SEC2:EnsembleDeCleints:=[Sec2_o,sec2_sec1,sec2_sec2,sec2_tma2];
    SEC3:EnsembleDeCleints:=[Sec3_o,sec3_sec1,sec3_sec2,sec3_tma3];
  end;
end;

procedure DONNER_LES_ALPHA(var T:TabClasse);
(* Cette proccédure permet de fournir au programmes les différentes
valeurs des alpha. Ces valeur sont enregistrées dans un tableau*)
var i:Client;
    c:integer;
begin
  writeln('Donner les taux d''arrivées de l''extérieur');
  writeln('DONNER DES VALEURS ENTIERES POSITIVES S.V.P');
  c:=1;
  for i:=emb1 to sec3_tma3 do
  begin
    if i in AvionExterieur then
    begin
      case c of
        1:write('alpha emb1 = ');

```



```

2:write('alpha emb2 = ');
3:write('alpha emb3 = ');
26:write('alpha sec1_sec2 = ');
27:write('alpha sec1_sec3 = ');
28:write('alpha sec1_tma1 = ');
30:write('alpha sec2_sec1 = ');
31:write('alpha sec2_sec3 = ');
32:write('alpha sec2_tma2 = ');
34:write('alpha sec3_sec1 = ');
35:write('alpha sec3_sec2 = ');
36:write('alpha sec3_tma3 = ');
end;
readln(T[i]);
c:=succ(c);
end
else
begin
T[i]:=0;
c:=succ(c);
end;
end;
end;

procedure DONNER_LES_MU(var T:TabClasse);
(* Cette proccédure permet de fournir au programmes les différétes
valeurs des mu. Ces valeurs sont enregistrées dans un tableau*)
var i:Client;
c:integer;
begin
writeln('Donner les taux du temps de service');
writeln('DONNER DES VALEURS ENTIERES, POSITIVES ET NONS NULLES
S.V.P');
c:=1;
for i:=emb1 to sec3_tma3 do
begin
case c of
1:write('mu emb1 = ');
2:write('mu emb2 = ');
3:write('mu emb3 = ');
4:write('mu deb1 = ');
5:write('mu deb2 = ');
6:write('mu deb3 = ');
7:write('mu twy1_gate1 = ');
8:write('mu twy1_rwy1 = ');
9:write('mu twy2_gate2 = ');
10:write('mu twy2_rwy2 = ');
11:write('mu twy3_gate3 = ');
12:write('mu twy3_rwy3 = ');
13:write('mu att1 = ');
14:write('mu dec1 = ');
15:write('mu att2 = ');
16:write('mu dec2 = ');
17:write('mu att3 = ');
18:write('mu dec3 = ');
19:write('mu tma1_rwy1 = ');
20:write('mu tma1_sec1 = ');
21:write('mu tma2_rwy2 = ');
22:write('mu tma2_sec2 = ');
23:write('mu tma3_rwy3 = ');
24:write('mu tma3_sec3 = ');

```

```

25:write('mu sec1_o = ');
26:write('mu sec1_sec2 = ');
27:write('mu sec1_sec3 = ');
28:write('mu sec1_tma1 = ');
29:write('mu sec2_o = ');
30:write('mu sec2_sec1 = ');
31:write('mu sec2_sec3 = ');
32:write('mu sec2_tma2 = ');
33:write('mu sec3_o = ');
34:write('mu sec3_sec1 = ');
35:write('mu sec3_sec2 = ');
36:write('mu sec3_tma3 = ');
end;(*fin case *)
readln(T[i]);
c:=succ(c);
end;(*fin boucle for*)
end;(*fin procedure*)

```

```

procedure DONNER_PROBA_TRANSITION(var T:MatriceProba);
(*Cette procédure permet de donner au pgm les probabilités de
transition*)
var i,j:Client;
    c:integer;(*Compteur pour afficher les elts du type énuméré*)
begin
    writeln('Donner les Probilités de transition');
    writeln('DONNER DES VALEURS ENTRE 0 et 1 INCLUS S.V.P');
    c:=1;
    for i:=emb1 to sec3_tma3 do

        for j:=emb1 to sec3_tma3 do
            begin
                case c of
                    109:begin
                        write('Proba de passer de deb1 à emb1 = ');
                        readln(T[i,j]);
                        c:=succ(c);
                    end;
                    116:begin
                        write('          de deb1 à twy1_rwy1 = ');
                        readln(T[i,j]);
                        c:=succ(c);
                    end;
                    146:begin
                        write('Proba de passer de deb2 à emb2 = ');
                        readln(T[i,j]);
                        c:=succ(c);
                    end;
                    155:begin
                        write('          de deb2 à twy2_rwy2 = ');
                        readln(T[i,j]);
                        c:=succ(c);
                    end;
                    183:begin
                        write('Proba de passer de deb3 à emb3 = ');
                        readln(T[i,j]);
                        c:=succ(c);
                    end;
                    192:begin
                        write('          de deb3 à twy3_rwy3 = ');
                        readln(T[i,j]);

```



```

        c:=succ(c);
    end;
217:begin
write('Proba de passer de twy1_gate1 à emb1 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
220:begin
write('                de twy1_gate1 à deb1 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
290:begin
write('Proba de passer de twy2_gate2 à emb2 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
293:begin
write('                de twy2_gate2 à deb2 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
363:begin
write('Proba de passer de twy3_gate3 à emb3 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
366:begin
write('                de twy3_gate3 à deb3 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
709:begin
write('Proba de passer de tma1_sec1 à sec1_o = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
710:begin
write('                de tma1_sec1 à sec1_sec2 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
711:begin
write('                de tma1_sec1 à sec1_sec3 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
785:begin
write('Proba de passer de tma2_sec2 à sec2_o = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
786:begin
write('                de tma2_sec2 à sec2_sec1 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
787:begin
write('                de tma2_sec2 à sec2_sec3 = ');
    readln(T[i,j]);

```

```

        c:=succ(c);
    end;
861:begin
write('Proba de passer de tma3_sec3 à sec3_o = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
862:begin
write('                de tma3_sec3 à sec3_sec1 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
863:begin
write('                de tma3_sec3 à sec3_sec2 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
929:begin
write('Proba de passer de sec1_sec2 à sec2_o = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
931:begin
write('                de sec1_sec2 à sec2_sec3 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
932:begin
write('                de sec1_sec2 à sec2_tma2 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
969:begin
write('Proba de passer de sec1_sec3 à sec3_o = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
971:begin
write('                de sec1_sec3 à sec3_sec2 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
972:begin
write('                de sec1_sec3 à sec3_tma3 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
1069:begin
write('Proba de passer de sec2_sec1 à sec1_o = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
1071:begin
write('                sec2_sec1 à sec1_sec3 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
1072:begin
write('                sec2_sec1 à sec1_tma1 = ');
    readln(T[i,j]);

```



```

        c:=succ(c);
    end;
1113:begin
write('Proba de passer de sec2_sec3 à sec3_o = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
1114:begin
write('                sec2_sec3 à sec3_sec1 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
1116:begin
write('                sec2_sec3 à sec3_tma3 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
1213:begin
write('Proba de passer de sec3_sec1 à sec1_o = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
1214:begin
write('                sec3_sec1 à sec1_sec2 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
1216:begin
write('                sec3_sec1 à sec1_tma1 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
1253:begin
write('Proba de passer de sec3_sec2 à sec2_o = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
1254:begin
write('                sec3_sec2 à sec2_sec1 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
1256:begin
write('                sec3_sec2 à sec2_tma2 = ');
    readln(T[i,j]);
    c:=succ(c);
    end;
8,46,84,266,340,414,439,
488,513,562,587,639,661,735,
809,991,1137,1283:
    begin
        T[i,j]:=1;
        c:=succ(c);
    end
else
    begin
        T[i,j]:=0;
        c:=succ(c)
    end;
end;
end;

```

```

end;

end;

procedure GARNIR_LES_LAMBDA(var T:TabClasseR);
begin
  T[emb1]:=1.2;
  T[emb2]:=2;
  T[emb3]:=1.5;
  T[deb1]:=1.3;
  T[deb2]:=2.5;
  T[deb3]:=1.2;
  T[twy1_gate1]:=1;
  T[twy1_rwy1]:=3;
  T[twy2_gate2]:=1.6;
  T[twy2_rwy2]:=2;
  T[twy3_gate3]:=1.4;
  T[twy3_rwy3]:=2.5;
  T[att1]:=2.3;
  T[dec1]:=3;
  T[att2]:=2;
  T[dec2]:=4;
  T[att3]:=2;
  T[dec3]:=1;
  T[tma1_rwy1]:=8;
  T[tma1_sec1]:=6;
  T[tma2_rwy2]:=5;
  T[tma2_sec2]:=6;
  T[tma3_rwy3]:=3;
  T[tma3_sec3]:=7;
  T[sec1_o]:=9;
  T[sec1_sec2]:=10;
  T[sec1_sec3]:=8;
  T[sec1_tma1]:=10;
  T[sec2_o]:=12;
  T[sec2_sec1]:=7;
  T[sec2_sec3]:=8;
  T[sec2_tma2]:=8;
  T[sec3_o]:=9;
  T[sec3_sec1]:=7;
  T[sec3_sec2]:=5;
  T[sec3_tma3]:=10.5;
end;(*fin procedure*)

procedure CALCULER_LES_RHOj(T1:TabClasseR;T2:TabClasse;var
T3:TabClasseR);
var j:Client;
begin
  for j:=emb1 to sec3_tma3 do
  begin
    T3[j]:=(T1[j]/T2[j])
  end;
end;

procedure CALCULER_LES_Vj_cm(T1:TabClasse;M:MatriceProba;var
T2:TabClasseR);
(*T1=tableau des mu. M=matrice des proba.T2=tableau des Vj_Cm *)
var j:Client;
begin
  for j:=emb1 to sec3_tma3 do

```



```

T2[j] := 1/T1[j];
T2[deb1] := ((1/T1[deb1]) + (M[deb1, emb1]/T1[emb1]));
T2[deb2] := ((1/T1[deb2]) + (M[deb2, emb2]/T1[emb2]));
T2[deb3] := ((1/T1[deb3]) + (M[deb3, emb3]/T1[emb3]));
end;

procedure CALCULER_LES_V_Classe(T1:TabClasse;var T2:TabClasseR);
(*T1=tableau des mu. M=matrice des proba.T2=tableau des Vj_Cm *)
var j:Client;
begin
  for j:=emb1 to sec3_tma3 do
    T2[j] := 1/T1[j];
  end;

function RHO_O_Cm(m:Serveur):real;
var j:Client;
    Temp:real;
    Cm:Avion;
begin
  DETERMINER_Cm(m, Cm);
  Temp:=0;
  for j:=emb1 to sec3_tma3 do
    if j in Cm then Temp:=Temp+Alpha[j]*V_Cm[j];
  RHO_O_Cm:=Temp;
end;

function RHO_S(S:Avion):real;
var c:Client;
    r:real;
begin
  r:=0;
  for c:=emb1 to sec3_tma3 do
    if c in S then r:=r+Rho_j[c];
  RHO_S:=r;
end;

function Ri_m(i:Client;m:Serveur):real;
(*i=classe de l'avion,m=station de service,Matrice=matrice des
proba,T1=mu,T2=Vj_Cm*)
var n:real;(*Variable qui contiendra le résultat provisoire de la
fonction*)
    Cm:Avion;
    j:Client;
begin
  DETERMINER_Cm(m, Cm);
  n:=0;
  for j:=emb1 to sec3_tma3 do
    if j in Cm then n:=n+Mu[i]*Proba[i,j]*V_Cm[j];
  Ri_m:=n;
end;

function R_BARRE_M1_M2(m1,m2:Serveur):real;
(*pic d'intensité de trafic de m1 vers m2*)
var Temp:real;
    Cm1:Avion;
    i:Client;
begin
  DETERMINER_Cm(m1, Cm1);
  Temp:=0;

```

```

    for i:=emb1 to sec3_tma3 do
        if i in Cm1 then
            if Temp < Ri_m(i,m2) then (*il faut des variables *)
                Temp:=Ri_m(i,m2);
            R_BARRE_M1_M2:=Temp;
        end;

function RHO_BARRE(m:Serveur):real;
    var Temp:real;
        j:Serveur;
        Diff:Station; (*set of Serveur*)
        Cm:Avion;
    begin
        DETERMINER_Cm(m,Cm);
        Diff:=StationsDeService-[m];
        Temp:=RHO_O_Cm(m);(*mettre les tableaux correspondants*)
        for j:=GATE1 to SEC3 do
            if j in Diff then Temp:=Temp+R_BARRE_M1_M2(j,m);
        RHO_BARRE:=Temp;
    end;

procedure GARNIR_RHO_Cm(var T:TabServeur);
    var s:Serveur;
        Cs:Avion;
    begin
        for s:=GATE1 to SEC3 do
            begin
                DETERMINER_Cm(s,Cs);
                T[s]:=RHO_S(Cs);
            end;
        end;

procedure GARNIR_RHO_BARRE_M(var T:TabServeur);
    var s:Serveur;
    begin
        for s:=GATE1 to SEC3 do
            T[s]:=RHO_BARRE(s);
        end;

function STABILITE_GLOBALE:boolean;
(*Les tableaux RhoBarreM et RhoCm doivent être garnis au préalable*)
    var s:Serveur;
        PasseLimite:boolean;
    begin
        s:=GATE1;
        PasseLimite:=false;
        while (s<=SEC3) and (not PasseLimite) do
            if (RhoBarreM[s]>=1) or (RhoCm[s]>=1) then
                PasseLimite:=true
            else
                s:=succ(s);
            STABILITE_GLOBALE:=not PasseLimite;
        end;

function BORNE_SUPERIEURE:real;
    var s:Serveur;
        j:Client;
        Temp:real;
        Cs:Avion;
    function MIN_V_Cm(S:Avion):real;

```



```

var c:Client;
Min:real;
begin
Min:=1000;
for c:=emb1 to sec3_tma3 do
if c in S then
if V_Cm[c]<Min then
Min:=V_Cm[c];
MIN_V_Cm:=Min;
end;
begin
Temp:=0;
for s:=GATE1 to SEC3 do
begin
DETERMINER_Cm(s,Cs);
for j:=emb1 to sec3_tma3 do
if j in Cs then
Temp:=Temp+(Rho_j[j]*V_Cm[j])/((1-
RhoBarreM[s])*MIN_V_Cm(Cs));
end;
BORNE_SUPERIEURE:=Temp;
end;

procedure KLIMOV(var Pi:TypePi;var GammaPi:TypeGammaPi;var
Y_BarreS:TypeGammaPi);
var CsurA:TabClasseR;
Min :real;
PosMin:Client;
k:integer;
cli:Client;
procedure RECHERCHER_MINIMUM(T:TabClasseR;var Minimum:real;var
Position:Client);
var i,j:Client;
Compteur:integer;
begin
Minimum:=2000;
for i:=emb1 to sec3_tma3 do
Compteur:=1;
for j:=emb1 to sec3_tma3 do
begin
if j in S[Compteur] then
if T[j]<Minimum then
begin
Minimum:=T[j];
Position:=j;
T[j]:=2000;
end;
Compteur:=succ(Compteur);
end;
end;
end;
procedure MAJ_TAB(var T:TabClasseR);
var i:Client;
Temp:real;
Compteur:integer;
begin
Temp:=0;
Compteur:=1;
for i:=emb1 to sec3_tma3 do
if i in S[k] then
begin

```

```

    Temp:=Temp+V_Cm[i]*Y_Barres[Compteur];
    CsurA[i]:=(Cout[i]-Temp)/V_Cm[i];
end;
Compteur:=succ(Compteur);
end;
begin
S[1]:=ClassesDAvions;
for Cli:=emb1 to sec3_tma3 do
    CsurA[Cli]:=Cout[Cli]/V_Cm[Cli];
RECHERCHER_MINIMUM(CsurA,Min,PosMin);
Y_Barres[1]:=Min;
GammaPi[1]:=Y_Barres[1];
Pi[1]:=PosMin;
for k:=2 to N do
    begin
        S[k]:=S[k-1]-[PosMin];
        MAJ_TAB(CsurA);
        RECHERCHER_MINIMUM(CsurA,Min,PosMin);
        Y_Barres[k]:=Min;
        GammaPi[k]:=GammaPi[k-1]+Y_Barres[k];
        (*Terminaison de l'algo??? Step N+1 *)
    end;
end;

function
R_M1_M2(Server1:Serveur;Server2:Serveur;Classe:Client;Ensemble:Avion):real;
var Cm1,Cm2:Avion;
    Temp:real;
    k:Client;
begin
    DETERMINER_Cm(Server1,Cm1);
    DETERMINER_Cm(Server2,Cm2);
    Temp:=0;
    for k:=emb1 to sec3_tma3 do
        if k in Ensemble*Cm2 then
            Temp:=Temp+Mu[k]*Proba[Classe,k]*V_Cm[k];
    R_M1_M2:=Temp;
end;

function B_CHAPEAU_DE_S_INTER_Cm(Ensemble:avion;Server:Serveur):real;
var Intersection:avion;
    Cm:Avion;
    Temp:real;
    Cli:Client;
begin
    DETERMINER_Cm(Server,Cm);
    Intersection:=Ensemble*Cm;
    if Intersection=Cm then
        for Cli:=emb1 to sec3_tma3 do
            if Cli in Intersection then
                Temp:=Temp+(Rho_j[Cli]*V_Cm[Cli])/(1-
RHO_O_Cm(Intersection))
            else
        else
            if Intersection=[] then
                Temp:=Temp+0
            else
                Temp:=(Temp+Rho_j[Cli]*V_Classe[Cli])/(1-
RHO_O_S(Intersection));

```



end;

{ \$M 10000,40000 }  
begin

end.